

Mesodinium chamaeleon data analysis

Holly Moeller

August 20, 2020

Loading of packages and data

```
# LOAD RELEVANT PACKAGES
require(Rmisc) #has summarySE function, which is quite useful

## Loading required package: Rmisc
## Loading required package: lattice
## Loading required package: plyr
require(sciplot) #has bargraph.CI function

## Loading required package: sciplot
require(pracma)

## Loading required package: pracma

# IMPORT DATA
setwd("~/GoogleSync/KarmaChamaeleon/MC_DataAnalysis") # set working directory
dat<-read.csv('Meso_TS_v2.csv',header=TRUE) # Timeseries of M. chamaeleon data
crypto.dat <- read.csv('Crypto_TS.csv',header=TRUE) # Timeseries of cryptophyte data
growth<-read.csv('GrowthRates.csv',header=TRUE) # Growthrates file
pe <- read.csv('MC_PEcureData.csv',header=TRUE) # M. chamaeleon photosynthesis-irradiance data
pe.crypto <- read.csv('PEData_cryptos.csv',header=TRUE) # Cryptophyte photosynthesis-irradiance data

# Adjust necessary data into numeric form; create ID columns
str(dat)

## 'data.frame':    648 obs. of  39 variables:
## $ Expt          : int  NA NA NA NA NA NA NA NA NA NA NA ...
## $ MCstrain      : int  1802 1802 1802 1802 1802 1802 1802 1802 1802 1802 1802 ...
## $ Prey          : Factor w/ 4 levels "CM","HP","RS",...: 3 3 3 3 3 3 3 3 3 3 3 ...
## $ Light         : int  100 100 100 100 100 100 100 100 100 100 100 ...
## $ Fed           : Factor w/ 2 levels "N","Y": 1 1 1 1 1 1 1 1 1 1 1 ...
## $ Rep           : Factor w/ 3 levels "A","B","C": 1 1 1 1 1 1 1 1 1 1 2 ...
## $ ExptDay       : int  0 1 2 4 6 8 10 12 14 0 ...
## $ MC_cellspmL   : num  344 450 925 1069 1393 ...
## $ Prey_cellspmL : num  0 0 0 0 0 0 0 0 0 0 ...
## $ corr_chl_p_cell : Factor w/ 640 levels "#VALUE!","10.00180734",...: 278 200 13 600 467 445 417 ...
## $ FIREtimestamp : int  58583 56430 50690 51462 51235 79406 59703 55264 62212 59770 ...
## $ FvFm          : num  0.378 0.326 0.341 0.3 0.224 0.21 0.192 0.2 0.149 0.262 ...
## $ Fm            : int  5336 4075 3961 5892 5653 4680 6898 5134 7046 7345 ...
## $ Pmax          : num  101.6 146.5 127.4 126.3 91.4 ...
## $ a             : num  1.12 1.065 1.184 1.408 0.877 ...
## $ Pmax.err      : num  NA NA NA NA NA NA NA NA NA NA NA ...
## $ Pmax.pvalue   : num  NA NA NA NA NA NA NA NA NA NA NA ...
## $ a.err         : num  NA NA NA NA NA NA NA NA NA NA NA ...
## $ a.pvalue      : num  NA NA NA NA NA NA NA NA NA NA NA ...
```

```

## $ E_K           : num  90.7 137.6 107.6 89.7 104.3 ...
## $ ETR_I         : num  81.4 91 93 101.8 68 ...
## $ Ct_crypto     : num  34.1 33.9 31.7 32.9 31.8 ...
## $ crypto_qPCR_run : Factor w/ 9 levels "", "10Aug_run1",...: 9 9 9 9 9 9 9 9 9 ...
## $ crypto_count   : num  686 743 2209 1222 2165 ...
## $ Ct_meso       : num  29.3 29.5 28 28.8 28.1 ...
## $ meso_qPCR_run  : Factor w/ 9 levels "", "10Aug_run1",...: 6 6 6 6 6 6 6 6 6 ...
## $ meso_count     : Factor w/ 413 levels "#VALUE!", "10.21692017",...: 394 368 153 48 138 108 211
## $ est_meso       : num  688 900 1850 2138 2786 ...
## $ Ct_rawratio    : Factor w/ 645 levels "", "#VALUE!", "1.032245532",...: 481 404 286 350 270 379
## $ PlastidCount   : Factor w/ 642 levels "", "#VALUE!", "0",...: 542 633 624 623 48 579 638 341 52
## $ cryptos_per_est_meso: Factor w/ 646 levels "", "#DIV/0!", "0.00000",...: 93 80 110 56 75 51 109 98 4
## $ X              : logi  NA NA NA NA NA NA ...
## $ X.1            : logi  NA NA NA NA NA NA ...
## $ X.2            : logi  NA NA NA NA NA NA ...
## $ X.3            : logi  NA NA NA NA NA NA ...
## $ X.4            : logi  NA NA NA NA NA NA ...
## $ X.5            : logi  NA NA NA NA NA NA ...
## $ X.6            : logi  NA NA NA NA NA NA ...
## $ X.7            : logi  NA NA NA NA NA NA ...

dat$ID <- paste(dat$Prey, dat$Light, dat$Fed, dat$Rep, sep='.')
dat$Pmax_chl <- dat$Pmax*0.06332817 # The multiplier 0.06332817 converts from electrons per chlorophyll
dat$Pmax_cell <- dat$Pmax_chl*as.numeric(as.character(dat$corr_chl_p_cell))

## Warning: NAs introduced by coercion

dat$PI_chl <- dat$ETR_I*0.06332817
dat$PI_cell <- dat$PI_chl*as.numeric(as.character(dat$corr_chl_p_cell))

## Warning: NAs introduced by coercion

dat$chl_p_cell <- as.numeric(as.character(dat$corr_chl_p_cell)) # Sometimes R incorrectly imports data

## Warning: NAs introduced by coercion

dat$chl_p_mL <- dat$chl_p_cell*dat$MC_cellspmL/1000
dat$PlastidCount <- as.numeric(as.character(dat$PlastidCount))

## Warning: NAs introduced by coercion

str(crypto.dat)

## 'data.frame': 324 obs. of 28 variables:
## $ Expt          : logi  NA NA NA NA NA NA ...
## $ MCstrain      : Factor w/ 1 level "Ctrl": 1 1 1 1 1 1 1 1 1 ...
## $ Prey          : Factor w/ 4 levels "CM", "HP", "RS",...: 4 4 4 4 4 4 4 4 4 ...
## $ Light         : int   100 100 100 100 100 100 100 100 100 ...
## $ Fed           : logi  NA NA NA NA NA NA ...
## $ Rep           : Factor w/ 3 levels "A", "B", "C": 1 1 1 1 1 1 1 1 2 ...
## $ ExptDay       : int    0 1 2 4 6 8 10 12 14 0 ...
## $ MeanCells     : num   155 39 81 58 66.2 ...
## $ CountVol      : num    20 2 2 0.3 0.3 0.3 0.3 NA 0.1 20 ...
## $ CF            : num     1 1 1 0.6 0.6 0.6 0.6 NA 1 1 ...
## $ CellCount     : num   7750 19500 40500 116796 133241 ...
## $ ChlExtr       : num    19.9 47.2 110.4 210.8 76.9 ...
## $ ChlVol        : num     4 4 4 4 2 2 2 NA 2 4 ...
## $ ChlpmL        : num    24.9 59.1 138 263.4 192.3 ...

```

```
crypto.dat$ID <- paste(crypto.dat$Prey, crypto.dat$Light, crypto.dat$Rep, sep='.')
crypto.dat$Pmax_chl <- crypto.dat$Pmax*0.06332817
crypto.dat$Pmax_cell <- crypto.dat$Pmax_chl*crypto.dat$ChlpCell
crypto.dat$PI_chl <- crypto.dat$P_I*0.06332817
crypto.dat$PI_cell <- crypto.dat$PI_chl*crypto.dat$ChlpCell

pe$ETR.C <- pe$ETR*0.06332817 # The multiplier 0.06332817 converts from electrons per chlorophyll to mg C/mg chl/h

# SET PLOTTING PARAMETERS (COLOURS)
Acol <- 'red' # colour to plot rep A data
Bcol <- 'blue' # colour to plot rep B data
Ccol <- 'green' # colour to plot rep C data
Fedcol <- 'black'
Starvedcol <- 'gray80'
Cryptocol <- 'gray50'

SMcol <- 'firebrick'
RScol <- 'lightcoral'
CMcol <- 'seagreen'
HPcol <- 'turquoise3'
preycolvec <- c(CMcol, HPcol, RScol, SMcol)

days <- unique(dat$ExptDay)
```

```
# Pick max crypto response

crypto.max <- as.data.frame(unique(crypto.dat$ID))
colnames(crypto.max) <- c('ID')
crypto.max$Prey <- crypto.dat$Prey[match(crypto.max$ID,crypto.dat$ID)]
crypto.max$Light <- crypto.dat$Light[match(crypto.max$ID,crypto.dat$ID)]
crypto.max$Rep <- crypto.dat$Rep[match(crypto.max$ID,crypto.dat$ID)]
crypto.max$chl <- NaN; crypto.max$FvFm <- NaN; crypto.max$P_I <- NaN; crypto.max$a <- NaN;

for(i in 1:length(unique(crypto.dat$ID))){
  hold <- crypto.dat[crypto.dat$ID==unique(crypto.dat$ID)[i],]
```

```

crypto.max$chl[i] <- max(hold$ChlpCell,na.rm=TRUE)
hold2 <- hold
if(hold$Prey[1]=='CM'){ hold2 <- hold[hold$ExptDay%in%c(0,1),]}
crypto.max$FvFm[i] <- max(hold2$Fv.Fm,na.rm=TRUE)
hold <- hold[hold$ExptDay%in%c(0,1,2,4),]
crypto.max$P_I[i] <- max(hold$P_I,na.rm=TRUE)
crypto.max$Pmax[i] <- max(hold$Pmax,na.rm=TRUE)
crypto.max$a[i] <- max(hold$a,na.rm=TRUE)
crypto.max$Pmax_chl[i] <- max(hold$Pmax_chl,na.rm=TRUE)
crypto.max$Pmax_cell[i] <- max(hold$Pmax_cell,na.rm=TRUE)
crypto.max$PI_chl[i] <- max(hold$PI_chl,na.rm=TRUE)
crypto.max$PI_cell[i] <- max(hold$PI_cell,na.rm=TRUE)
}

MC.max <- as.data.frame(unique(dat$ID))
colnames(MC.max) <- c('ID')
MC.max$Prey <- dat$Prey[match(MC.max$ID,dat$ID)]
MC.max$Light <- dat$Light[match(MC.max$ID,dat$ID)]
MC.max$Fed <- dat$Fed[match(MC.max$ID,dat$ID)]
MC.max$Rep <- dat$Rep[match(MC.max$ID,dat$ID)]
MC.max$chl <- NaN; MC.max$FvFm <- NaN; MC.max$P_I <- NaN; MC.max$a <- NaN; MC.max$Pmax <- NaN; MC.max$Pmax_chl <- NaN; MC.max$Pmax_cell <- NaN; MC.max$PI_chl <- NaN; MC.max$PI_cell <- NaN
MC.max$plastids <- NaN

for(i in 1:length(unique(dat$ID))){
  hold <- dat[dat$ID==unique(dat$ID)[i],]
  hold <- hold[hold$ExptDay%in%c(0,1,2,4,6),]
  MC.max$P_I[i] <- max(hold$ETR_I,na.rm=TRUE)
  MC.max$a[i] <- max(hold$a,na.rm=TRUE)
  MC.max$Pmax[i] <- max(hold$Pmax,na.rm=TRUE)
  MC.max$Pmax_chl[i] <- max(hold$Pmax_chl,na.rm=TRUE)
  MC.max$Pmax_cell[i] <- max(hold$Pmax_cell,na.rm=TRUE)
  MC.max$PI_chl[i] <- max(hold$PI_chl,na.rm=TRUE)
  MC.max$PI_cell[i] <- max(hold$PI_cell,na.rm=TRUE)
  hold <- hold[hold$ExptDay%in%c(0,1,2,4),]
  MC.max$chl[i] <- max(hold$chl_p_cell,na.rm=TRUE)
  MC.max$FvFm[i] <- max(hold$FvFm,na.rm=TRUE)
  MC.max$plastids[i] <- max(hold$PlastidCount,na.rm=TRUE)
}

crypto.max$Prey.Light <- paste(crypto.max$Prey,crypto.max$Light,sep='.')
MC.max$Prey.Light <- paste(MC.max$Prey,MC.max$Light,sep='.')
MC.max$Treat.Rep <- paste('1802',MC.max$ID,sep='.')

MC.max$relFvFm <- NaN
MC.max$relP_I <- NaN
MC.max$rela <- NaN
MC.max$relPmax <- NaN
MC.max$relPmax_chl <- NaN
MC.max$relPI_chl <- NaN

for(i in 1:dim(MC.max)[1]){
  # Compute relative MC FuFm

```

```

callID <- MC.max$Prey.Light[i]
MC.max$relFvFm[i] <- MC.max$FvFm[i]/mean(crypto.max[crypto.max$Prey.Light==callID,]$FvFm,na.rm=TRUE)

# Compute relative MC P_I
MC.max$relP_I[i] <- MC.max$P_I[i]/mean(crypto.max[crypto.max$Prey.Light==callID,]$P_I,na.rm=TRUE)

# Compute relative MC a
MC.max$rela[i] <- MC.max$a[i]/mean(crypto.max[crypto.max$Prey.Light==callID,]$a,na.rm=TRUE)

# Compute relative MC P_max
MC.max$relPmax[i] <- MC.max$Pmax[i]/mean(crypto.max[crypto.max$Prey.Light==callID,]$Pmax,na.rm=TRUE)

# Compute relative MC P_max per chl
MC.max$relPmax_chl[i] <- MC.max$Pmax_chl[i]/mean(crypto.max[crypto.max$Prey.Light==callID,]$Pmax_chl,na.rm=TRUE)

# Compute relative MC P_I per chl
MC.max$relPI_chl[i] <- MC.max$PI_chl[i]/mean(crypto.max[crypto.max$Prey.Light==callID,]$PI_chl,na.rm=TRUE)
}

dat$Treat.Rep <- paste(dat$MCstrain,dat$Prey,dat$Light,dat$Fed,dat$Rep,sep='.')
#treat.data <- as.data.frame(unique(dat$Treatment))
growth$Treat.Rep <- paste(growth$MCstrain,growth$Prey,growth$Light,growth$Fed,growth$Rep,sep='.')

growth$ic <- NaN
growth$preyic <- NaN
growth$max <- NaN
growth$relincrease <- NaN
growth$relincreaseperprey <- NaN
for(i in 1:dim(growth)[1]){
  dat.hold <- dat[dat$Treat.Rep==growth$Treat.Rep[i],]
  growth$ic[i] <- dat.hold[dat.hold$ExptDay==0,]$MC_cellspmL
  growth$preyic[i] <- dat.hold[dat.hold$ExptDay==0,]$Prey_cellspmL
  growth$max[i] <- max(dat.hold$MC_cellspmL,na.rm=TRUE)
  growth$relincrease[i] <- growth$max[i]/growth$ic[i]
  growth$relincreaseperprey[i] <- (growth$max[i]-growth$ic[i])/growth$preyic[i]
}

growth$deltarelincreasefed <- NaN
for(i in 1:dim(growth[growth$Fed=='Y',])[1]){
  growth[growth$Fed=='Y',]$deltarelincreasefed[i] <- growth[growth$Fed=='Y',]$relincrease[i]-growth[growth$Fed=='Y',]$relincrease[i-1]
}

growth$TotPreyCons <- rep(0,dim(growth)[1])
times <- c(0,1,2,4,6,8,10,12,14)

for(i in 1:length(unique(dat$Treat.Rep))){
  callID <- unique(dat$Treat.Rep)[i] # Select the dataset of interest
  if(growth[growth$Treat.Rep==callID,]$Fed=='Y'){ # If this is a fed dataset
    hold <- dat[dat$Treat.Rep==callID,]

```

```

    hold <- hold[!is.na(hold$MC_cellspmL),]
    prey <- hold$Prey_cellspmL
    MC <- hold$MC_cellspmL
    MC_init <- MC[1]
    graz <- growth[growth$Treat.Rep==callID,]$g
    fcn <- graz*prey
    for(j in 1:length(hold)[1]){if(is.nan(fcn[j])){fcn[j]<-0}}
    growth[growth$Treat.Rep==callID,]$TotPreyCons <- trapz(hold$ExptDay,fcn)
    growth[growth$Treat.Rep==callID,]$MC_init <- MC_init
  }
}

growth$PreyperMC <- growth$TotPreyCons/growth$ic

# photosynthesis contributions

growth$chl.loss.rate <- NaN
growth$plastid.loss.rate <- NaN
growth$FvFm.decline <- NaN
growth$chl.decline <- NaN
growth$FvFm.pctdecline <- NaN
growth$chl.pctdecline <- NaN
growth$PI.perchl <- NaN
growth$Cfixforchl <- NaN
growth$chl.res.time <- NaN
growth$C.perchl <- NaN
growth$C.perprey <- NaN
growth$C.fromprey <- NaN
growth$C.fixed <- NaN
growth$C.fixed2 <- NaN

IDs <- unique(growth$Treat.Rep)
times <- c(0,1,2,4,6,8,10,12,14)
tsteps <- 5 # number of timesteps to use

for(i in 1:length(IDs)){
  # Choose the dataset
  callID <- IDs[i]
  hold <- dat[dat$Treat.Rep==callID,]

  # Determine rate of chlorophyll loss over first week
  chls <- log(hold$chl_p_cell)
  lm1 <- lm(chls[1:tsteps]~times[1:tsteps])
  if(hold$Prey[1]=="CM"){ lm1 <- lm(chls[2:tsteps]~times[2:tsteps])
  if(hold$Light[1]==10&hold$Rep[1]=='B'){ lm1 <- lm(chls[1:4]~times[1:4]) }}

  growth[growth$Treat.Rep==callID,]$chl.loss.rate <- summary(lm1)$coefficients[2,1]

  # Determine rate of plastid loss over first 48 hrs
  plastids <- log(hold$PlastidCount)
  lm2 <- lm(plastids[1:3]~times[1:3])
  growth[growth$Treat.Rep==callID,]$plastid.loss.rate <- summary(lm2)$coefficients[2,1]

```

```

# Determine overall decline in Fv/Fm
FvFm.init <- hold$FvFm[1]
FvFm.min <- min(hold$FvFm,na.rm=TRUE)
growth[growth$Treat.Rep==callID,]$FvFm.decline <- FvFm.init-FvFm.min
growth[growth$Treat.Rep==callID,]$FvFm.pctdecline <- (FvFm.init-FvFm.min)/FvFm.init

# Determine overall decline in chl-a
chla.init <- hold$chl_p_cell[1]
chla.min <- min(hold$chl_p_cell,na.rm=TRUE)
growth[growth$Treat.Rep==callID,]$chl.decline <- chla.init-chla.min
growth[growth$Treat.Rep==callID,]$chl.pctdecline <- (chla.init-chla.min)/chla.init

# Chlorophyll residence time
growth[growth$Treat.Rep==callID,]$chl.res.time = -1/(growth[growth$Treat.Rep==callID,]$chl.loss.rate)

# Carbon fixed during that residence time
PIs <- hold$PI_chl
PI.mean <- mean(PIs[1:tsteps],na.rm=TRUE)
growth[growth$Treat.Rep==callID,]$PI.perchl <- PI.mean
growth[growth$Treat.Rep==callID,]$C.perchl <- PI.mean*growth[growth$Treat.Rep==callID,]$chl.res.time

# Carbon fixed per prey cell
preychl <- mean(crypto.max[crypto.max$Prey==hold$Prey[1]&crypto.max$Light==hold$Light[1],]$chl,na.rm=TRUE)
growth[growth$Treat.Rep==callID,]$C.perprey <- preychl*growth[growth$Treat.Rep==callID,]$C.perchl

# Carbon fixed from all prey across the experiment
preycons <- mean(growth[growth$Fed=='Y'&growth$Prey==hold$Prey[1]&growth$Light==hold$Light[1],]$Prey)
growth[growth$Treat.Rep==callID,]$C.fromprey <- preycons*growth[growth$Treat.Rep==callID,]$C.perprey

# Carbon fixed, calculated as integral of chl-a/cell * PI_chl
hold <- hold[!is.na(hold$PI_cell),]
growth[growth$Treat.Rep==callID,]$C.fixed <- trapz(hold$ExptDay,hold$chl_p_cell*hold$PI_chl*12) # m

# Carbon fixed, calculated as integral of chl-a/mL * PI_chl divided by initial cells
hold <- hold[!is.na(hold$chl_p_mL),]
growth[growth$Treat.Rep==callID,]$C.fixed2 <- trapz(hold$ExptDay,hold$PI_chl*12*hold$chl_p_mL)/hold$chl_p_mL

}

growth$deltaCfixed <- NaN
for(i in 1:dim(growth[growth$Fed=='Y',,])[1]){
  growth[growth$Fed=='Y',]$deltaCfixed[i] <- growth[growth$Fed=='Y',]$C.fixed2[i]-growth[growth$Fed=='N',]$C.fixed2[i]
}

```

Visualizing Data

Figure 1: Loss of photosynthetic capacity in starved cultures

```

par(mar=c(2,4,0.5,0.5),mfrow=c(4,1))
xcoords <- c(1,2,3,4.5,5.5,6.5,8,9,10,11.5,12.5,13.5)

```

```

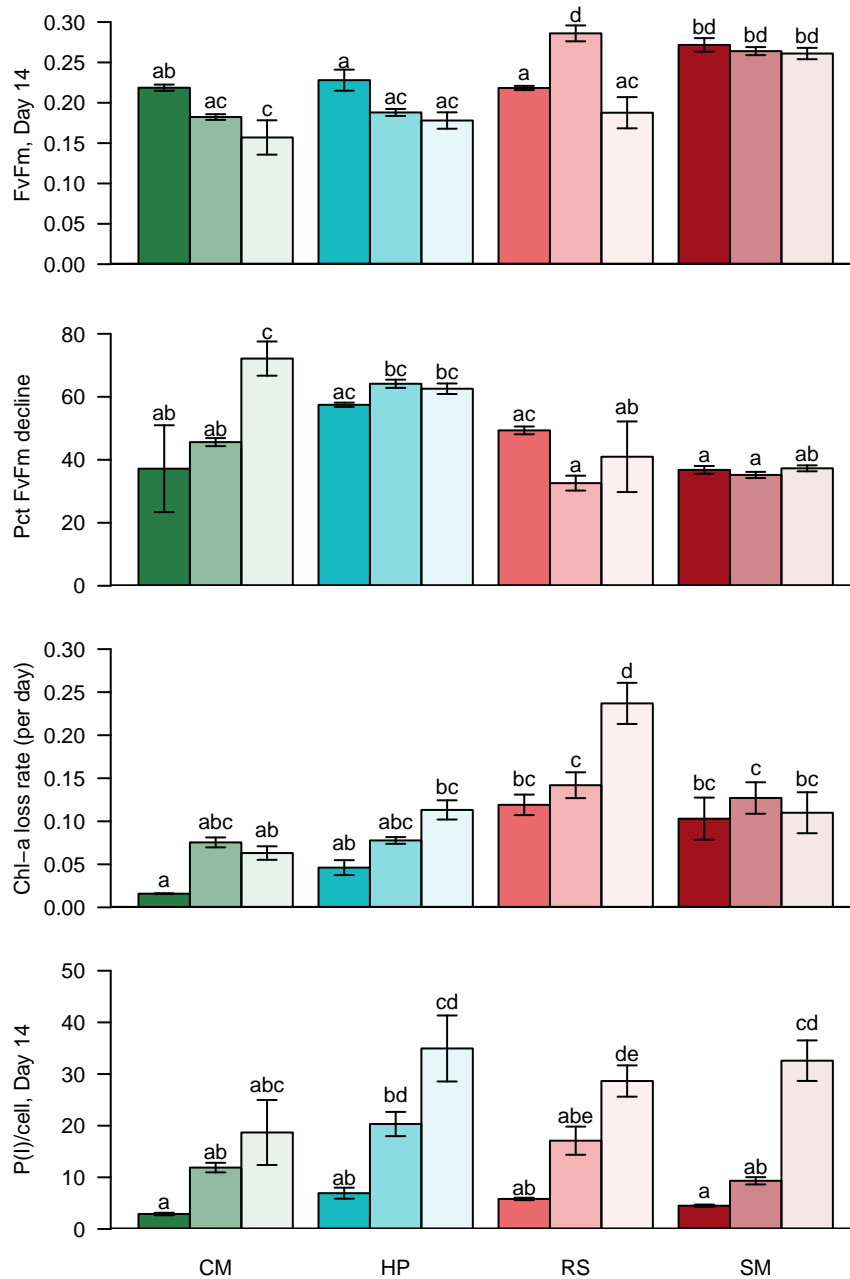
# FvFm at day 14
bargraph.CI(Prey,FvFm,group=Light,data=dat[dat$Fed=='N'&dat$ExptDay==14,],legend=FALSE,err.width=0.05,1)
#TukeyHSD(aov(FvFm~Prey*as.factor(Light),data=dat[dat$Fed=='N'&dat$ExptDay==14,]))

# decline in FvFm
#bargraph.CI(Prey,FvFm.decline,group=Light,data=growth[growth$Fed=='N',],legend=FALSE,err.width=0.05,la
bargraph.CI(Prey,100*FvFm.pctdecline,group=Light,data=growth[growth$Fed=='N',],legend=FALSE,err.width=0
#TukeyHSD(aov(FvFm.pctdecline~Prey*as.factor(Light),data=growth[growth$Fed=='N',]))

# pigment decay rate
bargraph.CI(Prey,-1*(chl.loss.rate),group=Light,data=growth[growth$Fed=='N',],legend=FALSE,err.width=0.
#TukeyHSD(aov(-1*(chl.loss.rate)~Prey*as.factor(Light),data=growth[growth$Fed=='N',]))

# P(I) at day 14
bargraph.CI(Prey,PI_cell,group=Light,data=dat[dat$Fed=='N'&dat$ExptDay==14,],legend=FALSE,err.width=0.0

```

```
#TukeyHSD(aov(PI_cell~Prey*as.factor(Light),data=dat[dat$Fed=='N'&dat$ExptDay==14,]))
```

Even after 14 days of starvation, *M. chamaeleon* acclimated to all four types showed sustained photosynthesis

Supplementary Figure: PE curves over time, by prey and light level, measured in electron transport per chlorophyll-a

```
# Note: this is in electron transport rate (electrons per s per chl-a)
```

```
# To convert to C, multiply by *0.06332817
```

```
pe.summ <- summarySE(data=pe, 'ETR', groupvars=c('Prey', 'Light', 'Fed', 'ExptDay', 'PAR'), na.rm=TRUE)
```

```

preychoice_set <- c('CM','HP','RS','SM')
lightchoice_set <- c(10,50,100)
ymax_set <- c(100,130,200)

days <- unique(pe.summ$ExptDay)
bgvec <- rev(c('gray90','gray80','gray70','gray60','gray50','gray40','gray30','gray20','black'))

par(mar=c(1,1,0.5,0.5))
layout(matrix(c(rep(c(13,1,1,1,2,2,2,3,3,3,4,4,4),3),rep(c(13,5,5,5,6,6,6,7,7,7,8,8,8),3),rep(c(13,9,9,9,10,10,10,11,11,11,12,12,12),3)),2,1))

## Warning in matrix(c(rep(c(13, 1, 1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4), 3),
## rep(c(13, : data length [133] is not a sub-multiple or multiple of the
## number of rows [10]

for(lightctr in 1:length(lightchoice_set)){
  lightchoice <- lightchoice_set[lightctr]
  ymaxchoice <- ymax_set[lightctr]

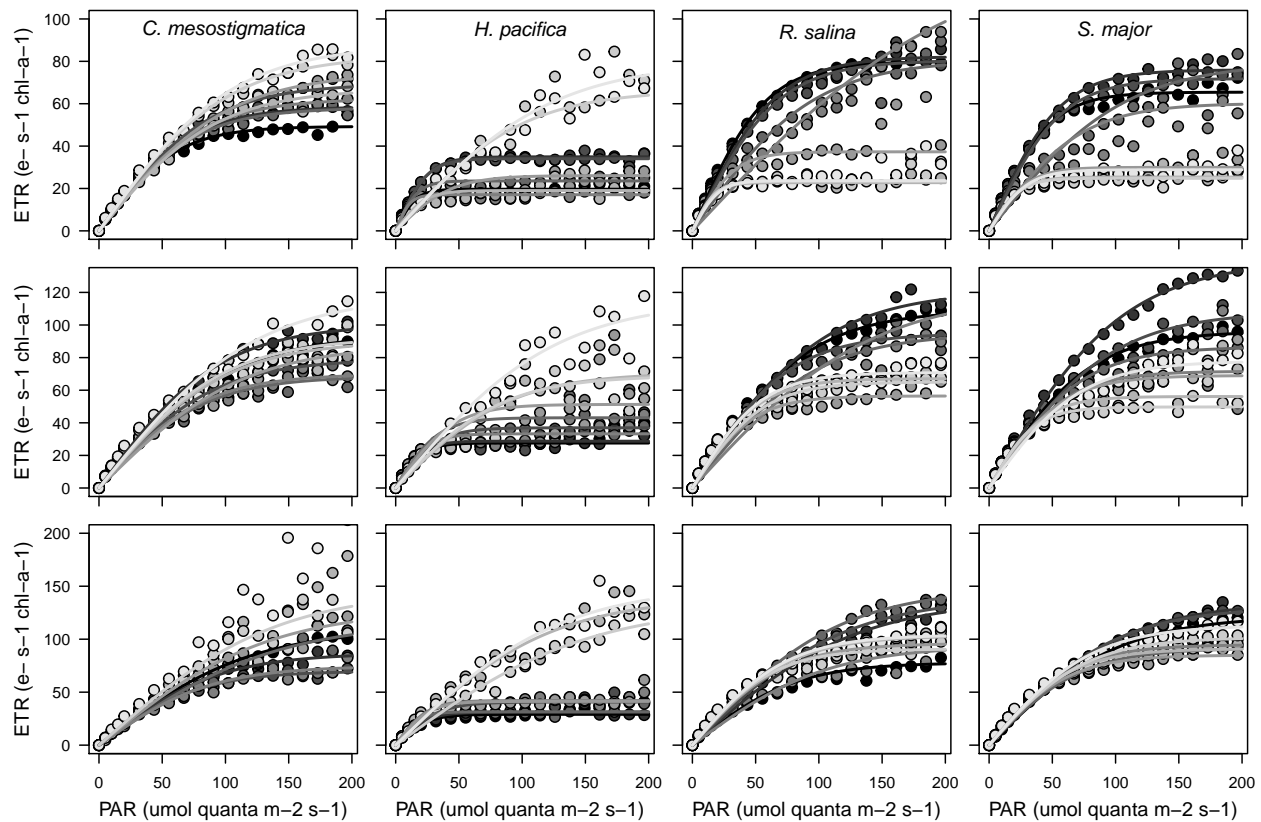
for(preycctr in 1:length(preychoice_set)){
  preychoice = preychoice_set[preycctr]

hold <- pe.summ[pe.summ$Prey==preychoice&pe.summ$Light==lightchoice&pe.summ$Fed=='N',]

plot(hold$PAR,hold$ETR,bg=bgvec[as.factor(hold$ExptDay)],pch=21,cex=1.5,xlab='',ylab='',las=1,ylim=c(0,
Iset <- seq(from = 0, to = 200,length.out=1000)
for(i in 1:length(days)){
  Pmax <- mean(dat[dat$Prey==preychoice&dat$Light==lightchoice&dat$Fed=='N'&dat$ExptDay==days[i],]$Pmax)
  alpha <- mean(dat[dat$Prey==preychoice&dat$Light==lightchoice&dat$Fed=='N'&dat$ExptDay==days[i],]$a)
  # Curves were fit with the Jassby & Platt hyperbolic tangent function:
  # ETR = P * tanh(a * PAR / P)

  ETRset <- Pmax*tanh(alpha*Iset/Pmax)
  lines(Iset,ETRset,col=bgvec[i],lwd=2)
}
if(lightctr==3){mtext('PAR (umol quanta m-2 s-1)',side=1,line=2.5,cex=0.8)}
if(preycctr==1){mtext('ETR (e- s-1 chl-a-1)',side=2,line=3,cex=0.8)}
if(lightctr==1){
  if(preycctr==1){mtext(substitute(paste(italic('C. mesostigmatica'))),side=3,line=-1.75,cex=.8)}
  if(preycctr==2){mtext(substitute(paste(italic('H. pacifica'))),side=3,line=-1.75,cex=.8)}
  if(preycctr==3){mtext(substitute(paste(italic('R. salina'))),side=3,line=-1.75,cex=.8)}
  if(preycctr==4){mtext(substitute(paste(italic('S. major'))),side=3,line=-1.75,cex=.8)}
}
}}

```



Supplementary Figure: PE curves over time, by prey and light level, measured in carbon fixation per cell

```

dat$Treat.Rep.Day <- paste(dat$Treat.Rep,dat$ExptDay,sep='.')
pe$Treat.Rep.Day <- paste(pe$MCstrain,pe$Prey,pe$Light,pe$Fed,pe$Rep,pe$ExptDay,sep='.')
pe$chl <- dat$corr_chl_p_cell[match(pe$Treat.Rep.Day,dat$Treat.Rep.Day)]
pe$ETR.cell <- pe$ETR.C*as.numeric(as.character(pe$chl))

## Warning: NAs introduced by coercion
pe.summ <- summarySE(data=pe,'ETR.cell',groupvars=c('Prey','Light','Fed','ExptDay','PAR'),na.rm=TRUE)

preychoice_set <- c('CM','HP','RS','SM')
lightchoice_set <- c(10,50,100)
ymax_set <- c(160,160,160)

days <- unique(pe.summ$ExptDay)
bgvec <- rev(c('gray90','gray80','gray70','gray60','gray50','gray40','gray30','gray20','black'))

par(mar=c(1,1,0.5,0.5))
layout(matrix(c(rep(c(13,1,1,1,2,2,2,3,3,3,4,4,4),3),rep(c(13,5,5,5,6,6,6,7,7,7,8,8,8),3),rep(c(13,9,9,9,10,10,10,11,11,11,12,12,12),3)),3,1))

## Warning in matrix(c(rep(c(13, 1, 1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4), 3),
## rep(c(13, : data length [133] is not a sub-multiple or multiple of the
## number of rows [10]

```

```

for(lightctr in 1:length(lightchoice_set)){
  lightchoice <- lightchoice_set[lightctr]
  ymaxchoice <- ymax_set[lightctr]

for(preycctr in 1:length(preychoice_set)){
  preychoice = preychoice_set[preycctr]

hold <- pe.summ[pe.summ$Prey==preychoice&pe.summ$Light==lightchoice&pe.summ$Fed=='N',]

plot(hold$PAR,hold$ETR.cell,bg=bgvec[as.factor(hold$ExptDay)],pch=21,cex=1.5,xlab='',ylab='',las=1,ylim=
Iset <- seq(from = 0, to = 200,length.out=1000)
for(i in 1:length(days)){
  Pmax <- mean(dat[dat$Prey==preychoice&dat$Light==lightchoice&dat$Fed=='N'&dat$ExptDay==days[i],]$Pmax)
  alpha <- mean(dat[dat$Prey==preychoice&dat$Light==lightchoice&dat$Fed=='N'&dat$ExptDay==days[i],]$a)*
  # Curves were fit with the Jassby & Platt hyperbolic tangent function:
  # ETR = P * tanh(a * PAR / P)

  ETRset <- Pmax*tanh(alpha*Iset/Pmax)
  lines(Iset,ETRset,col=bgvec[i],lwd=2)
}
if(lightctr==3){mtext('PAR (umol quanta m-2 s-1)',side=1,line=2.5,cex=0.8)}
if(preycctr==1){mtext('C fix (pg C cell-1 hr-1)',side=2,line=3,cex=0.8)}
if(lightctr==1){
  if(preycctr==1){mtext(substitute(paste(italic('C. mesostigmatica'))),side=3,line=-1.75,cex=.8)}
  if(preycctr==2){mtext(substitute(paste(italic('H. pacifica'))),side=3,line=-1.75,cex=.8)}
  if(preycctr==3){mtext(substitute(paste(italic('R. salina'))),side=3,line=-1.75,cex=.8)}
  if(preycctr==4){mtext(substitute(paste(italic('S. major'))),side=3,line=-1.75,cex=.8)}
}
}}

```

```

## Warning in mean(as.numeric(as.character(dat[dat$Prey == preychoice & dat
## $Light == : NAs introduced by coercion

## Warning in mean(as.numeric(as.character(dat[dat$Prey == preychoice & dat
## $Light == : NAs introduced by coercion

```

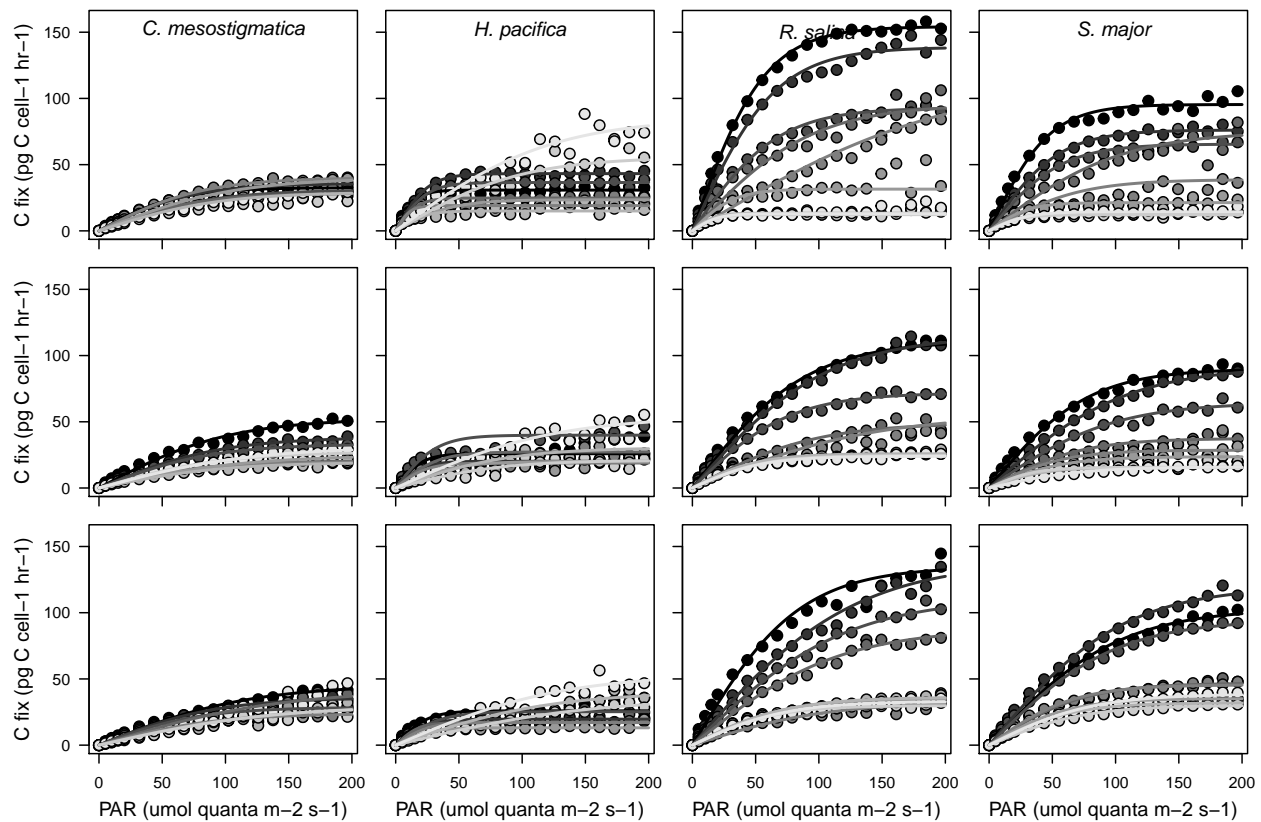


Figure 2: Maximum photosynthetic performance

```
#quartz(height=6.65,width=4)
xcoords <- c(1,2,3,4,5,5.5,6.5,8,9,10,11.5,12.5,13.5)
layout(matrix(c(1,1,2,2,3,3,4,4,5)))
par(mar=c(.5,4,0.5,2))

# Panel A: Fv/Fm
bargraph.CI(Prey,FvFm,group=as.factor(Light),data=MC.max[MC.max$Fed=='Y'],ylim=c(0.35,0.6),las=1,space=0)

#TukeyHSD(aov(FvFm~as.factor(Light)*Prey,data=MC.max[MC.max$Fed=='Y',]))

# Panel B: P_max per chl
bargraph.CI(Prey,Pmax_chl,group=as.factor(Light),data=MC.max[MC.max$Fed=='Y'],ylim=c(0,15),las=1,space=0)

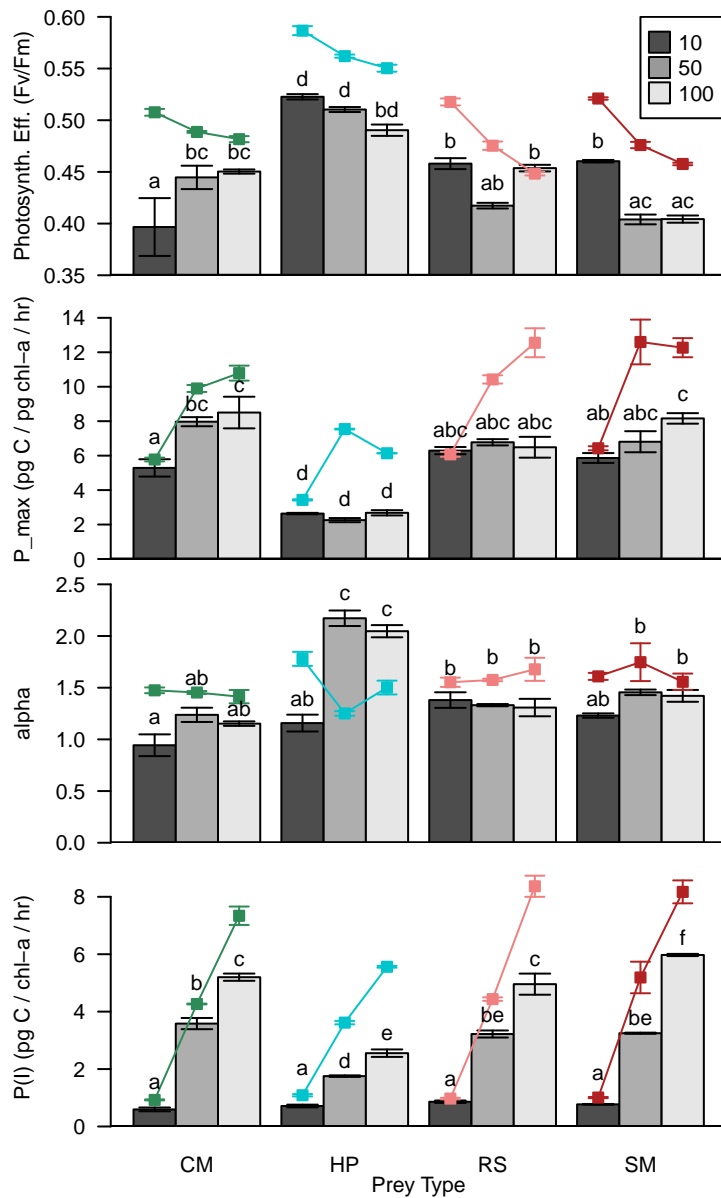
#TukeyHSD(aov(Pmax_chl~Prey*as.factor(Light),data=MC.max[MC.max$Fed=='Y',]))

# Panel C: alpha
bargraph.CI(Prey,a,group=as.factor(Light),data=MC.max[MC.max$Fed=='Y'],ylim=c(0,2.5),las=1,space=c(0,0))

#TukeyHSD(aov(a~Prey*as.factor(Light),data=MC.max[MC.max$Fed=='Y',]))

# Panel D: P(I) per chl
bargraph.CI(Prey,PI_chl,group=as.factor(Light),data=MC.max[MC.max$Fed=='Y'],ylim=c(0,9),space=c(0,0))

#TukeyHSD(aov(PI_chl~Prey*as.factor(Light),data=MC.max[MC.max$Fed=='Y',]))
```



Supplementary Figure: Relative photosynthetic capacity (compared to prey)

```
preytypes <- c('CM','HP','RS','SM'); ptcex <- 1.5
#quartz(width=6.28,height=5.44)
par(mar=c(4,4,0.2,1),mfrow=c(2,2))

plot(MC.max[MC.max$Fed=='Y'],$Light,MC.max[MC.max$Fed=='Y'],$relFvFm,pch=21,bg= preycolvec[MC.max[MC.max$Fed=='Y'],$Light])
plot(MC.max[MC.max$Fed=='Y'],$Light,MC.max[MC.max$Fed=='Y'],$relPmax_chl,pch=21,bg= preycolvec[MC.max[MC.max$Fed=='Y'],$Light])
plot(MC.max[MC.max$Fed=='Y'],$Light,MC.max[MC.max$Fed=='Y'],$relPmax_chl,pch=21,bg= preycolvec[MC.max[MC.max$Fed=='Y'],$Light])
plot(MC.max[MC.max$Fed=='Y'],$Light,MC.max[MC.max$Fed=='Y'],$relPI_chl,pch=21,bg= preycolvec[MC.max[MC.max$Fed=='Y'],$Light])
```

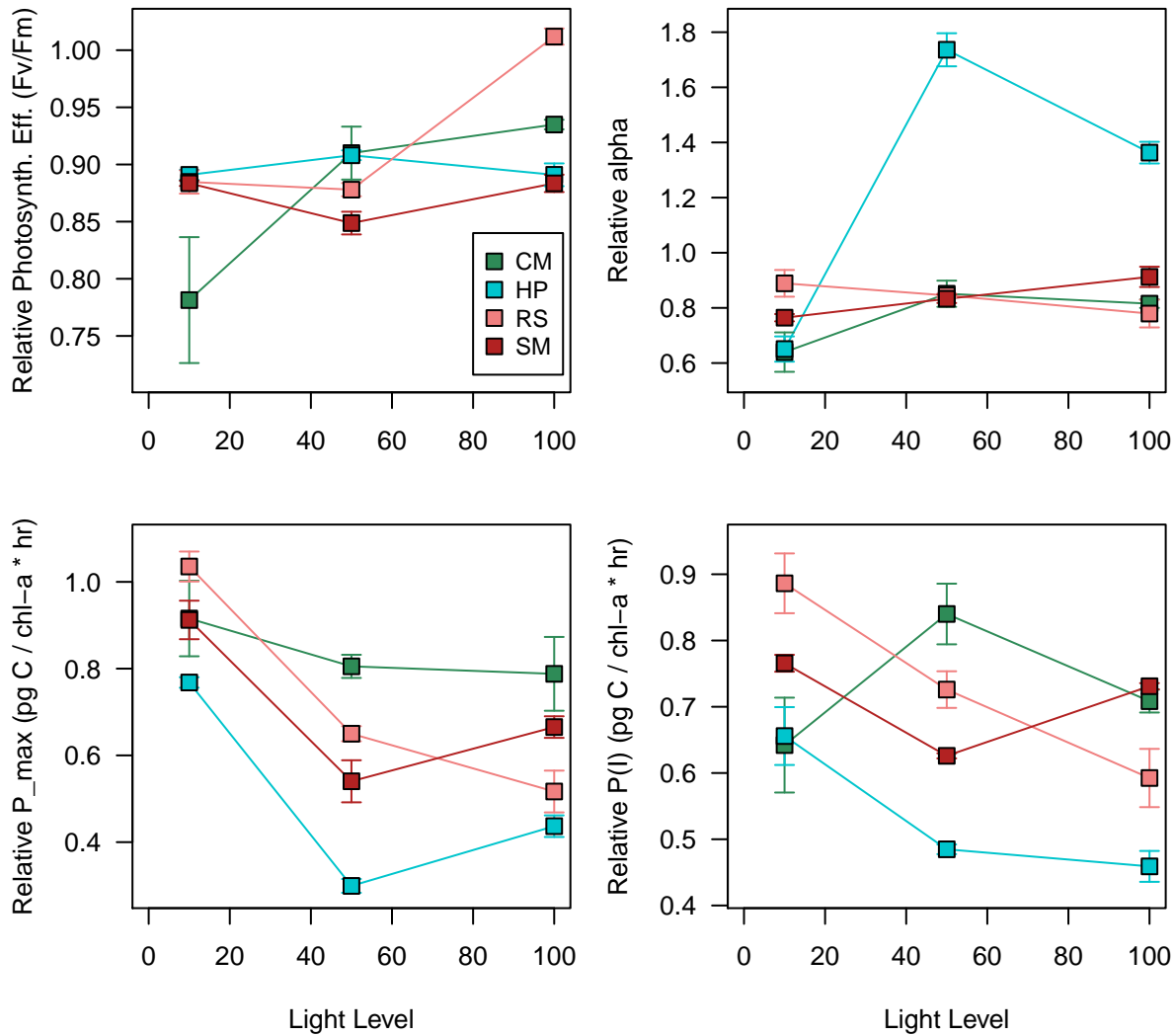


Figure 3: Per-cell plastids, pigment, and C fixation

```
#quartz(height=4.6,width=4.65)
xcoords <- c(1,2,3,4.5,5.5,6.5,8,9,10,11.5,12.5,13.5)
par(mar=c(0.5,4,0.5,4))
layout(matrix(c(1,1,1,2,2,2,3,3,3,4)))

bargraph.CI(Prey,plastids,group=Light,data=MC.max[MC.max$Fed=='Y'],legend=FALSE,x.leg=1,y.leg=15,ylim=
#TukeyHSD(aov(plastids~as.factor(Light)*Prey,data=MC.max[MC.max$Fed=='Y']))

# chl-a per cell
bargraph.CI(Prey,chl,group=as.factor(Light),data=MC.max[MC.max$Fed=='Y'],las=1,ylim=c(0,60),space=c(0,
legend(x = 0.25, y = 55, legend=c('10','50','100'), pch=22,pt.cex=2.5,pt.bg=c('gray30','gray60','gray90
#TukeyHSD(aov(chl~Prey*as.factor(Light),data=MC.max[MC.max$Fed=='Y']))

# P(I) per cell
```

```
bargraph.CI(Prey,PI_cell,group=as.factor(Light),data=MC.max[MC.max$Fed=='Y'],las=1,ylim=c(0,170),space=
mtext('Prey P(I) (pg C / crypto / hr)',side=4,line=2,cex=0.7); text(xcoords,c(18,108,118,24,43,43,45,9
#TukeyHSD(aov(PI_cell~Prey*as.factor(Light),data=MC.max[MC.max$Fed=='Y'],))
```

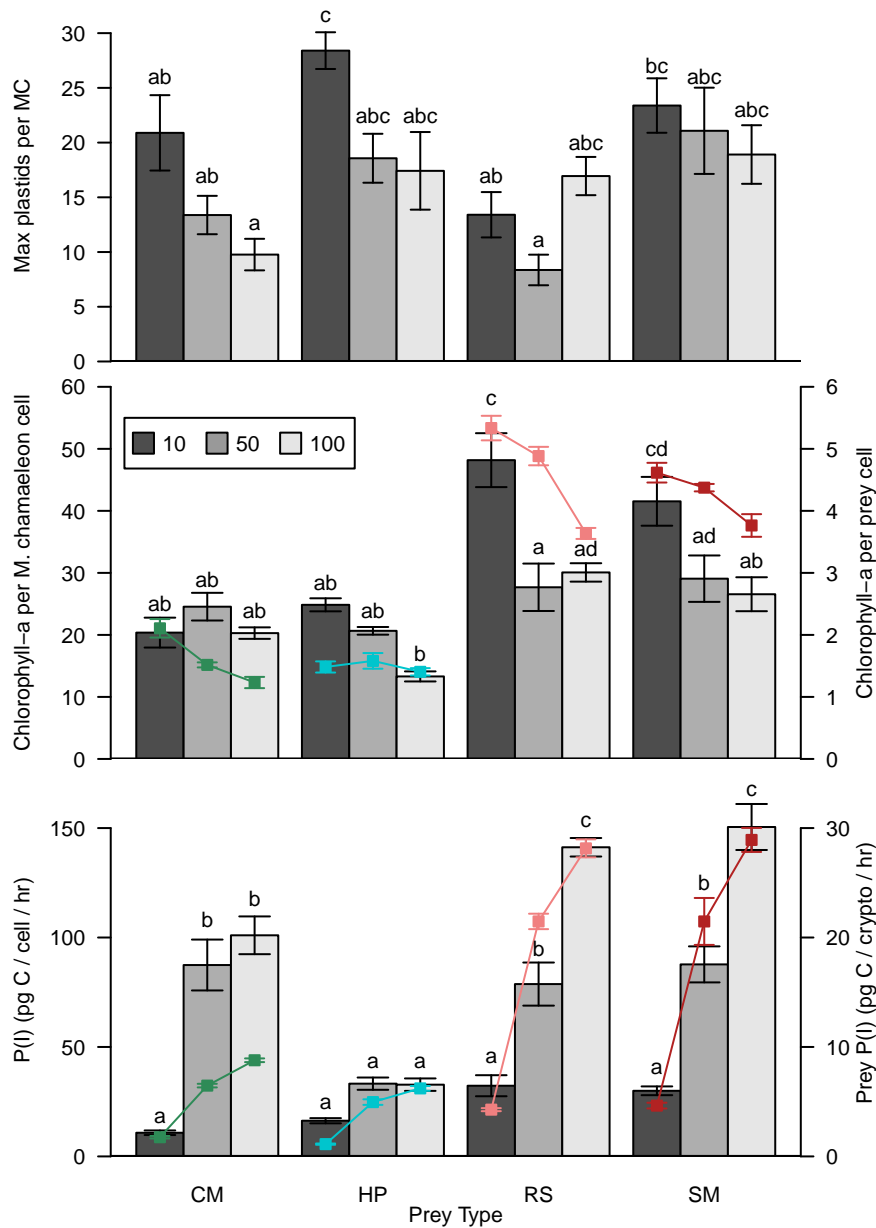


Figure 4: Growth rate and extent

```
growth.summ<-summarySE(data=growth,measurevar='muMC',groupvars=c('Prey','Light','Fed'))
growth$Prey.Light <- paste(growth$Prey,growth$Light,sep='.')
prey.growth.summ <- summarySE(data=growth[growth$Fed=='Y'],measurevar='muPreyCtrl',groupvars=c('Prey',
xcoords <- c(1.2,3.4,5.6)
xcoords2 <- c(.7,1.7,2.9,3.9,5.1,6.1)
```



```

#quartz(height=3.5,width=7.5)
par(mar=c(1,1,0.5,0.5))
layout(matrix(c(11,1,1,1,2,2,3,3,3,4,4,4,9,9,9,9,9,9,10,5,5,5,6,6,6,7,7,7,8,8,8,9,9,9,9,9),nrow=2,ncol=32),1)

# Column 1: Growth Rate
ymax <- 1.2; ymin <- -0.1; bargraph.CI(Light,muMC,group=Fed,data=growth[growth$Prey=='CM'],,las=1,legend=FALSE,leg.lab=c('Starved'))
bargraph.CI(Light,muMC,group=Fed,data=growth[growth$Prey=='HP'],,las=1,legend=FALSE,leg.lab=c('Starved'))
bargraph.CI(Light,muMC,group=Fed,data=growth[growth$Prey=='RS'],,las=1,legend=FALSE,leg.lab=c('Starved'))
bargraph.CI(Light,muMC,group=Fed,data=growth[growth$Prey=='SM'],,las=1,legend=FALSE,leg.lab=c('Starved'))

# Row 2: growth extent
xlabelstr <- '10uE          50uE          100uE'
ymax <- 12.4; ymin <- 0; bargraph.CI(Light,relincrease,group=Fed,data=growth[growth$Prey=='CM'],,las=1,legend=FALSE,leg.lab=c('Starved'))
bargraph.CI(Light, relincrease,group=Fed,data=growth[growth$Prey=='HP'],,las=1,legend=FALSE,leg.lab=c('Starved'))
bargraph.CI(Light, relincrease,group=Fed,data=growth[growth$Prey=='RS'],,las=1,legend=FALSE,leg.lab=c('Starved'))
bargraph.CI(Light, relincrease,group=Fed,data=growth[growth$Prey=='SM'],,las=1,legend=FALSE,leg.lab=c('Starved'))

par(mar=c(4,5,0.1,0.1))
ptcex <- 2

# First panel: growth rate vs. growth extent

lm.all <- lm(growth$relincrease~growth$muMC); summary(lm.all) #signif.; F-stat = 86.4, p-val = 7.616e-14

##
## Call:
## lm(formula = growth$relincrease ~ growth$muMC)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.8268 -1.0467 -0.4266  1.0790  5.4003
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.1771     0.4529   0.391   0.697
## growth$muMC  13.0391     1.4028   9.295 7.62e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.892 on 70 degrees of freedom
## Multiple R-squared:  0.5524, Adjusted R-squared:  0.546
## F-statistic: 86.4 on 1 and 70 DF, p-value: 7.616e-14

```

```

lm.RS <- lm(relincrease~muMC,data=growth[growth$Prey=='RS',]); summary(lm.RS) #signif.; F-stat = 39.13,

##
## Call:
## lm(formula = relincrease ~ muMC, data = growth[growth$Prey ==
##      "RS", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.1669 -1.3449  0.3205  1.3619  3.3652
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.942      1.262  -1.539   0.143
## muMC           21.457      3.430   6.255 1.15e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.951 on 16 degrees of freedom
## Multiple R-squared:  0.7098, Adjusted R-squared:  0.6916
## F-statistic: 39.13 on 1 and 16 DF,  p-value: 1.148e-05
lm.SM <- lm(relincrease~muMC,data=growth[growth$Prey=='SM',]); summary(lm.SM) #signif.; F-stat = 36.33,

##
## Call:
## lm(formula = relincrease ~ muMC, data = growth[growth$Prey ==
##      "SM", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.3789 -0.6495 -0.1635  0.4765  2.4325
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.1947      0.7310  -0.266   0.793
## muMC           13.2977      2.2061   6.028 1.76e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.194 on 16 degrees of freedom
## Multiple R-squared:  0.6943, Adjusted R-squared:  0.6752
## F-statistic: 36.33 on 1 and 16 DF,  p-value: 1.758e-05
lm.CM <- lm(relincrease~muMC,data=growth[growth$Prey=='CM',]); summary(lm.CM) #signif.; F-stat = 138.7,

##
## Call:
## lm(formula = relincrease ~ muMC, data = growth[growth$Prey ==
##      "CM", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.7871 -0.4654 -0.2081  0.5969  2.0592
##

```

```
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.9255     0.3480   2.659  0.0171 *
## muMC         16.1727     1.3733  11.777  2.7e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.054 on 16 degrees of freedom
## Multiple R-squared:  0.8966, Adjusted R-squared:  0.8901
## F-statistic: 138.7 on 1 and 16 DF,  p-value: 2.7e-09

lm.HP <- lm(relincrease~muMC,data=growth[growth$Prey=='HP',]); summary(lm.HP) #signif.; F-stat = 45.33,

##
## Call:
## lm(formula = relincrease ~ muMC, data = growth[growth$Prey ==
##       "HP", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82467 -0.28724  0.00287  0.15622  1.06984
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.5010     0.2923   1.714   0.106
## muMC         5.9977     0.8908   6.733 4.81e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5175 on 16 degrees of freedom
## Multiple R-squared:  0.7391, Adjusted R-squared:  0.7228
## F-statistic: 45.33 on 1 and 16 DF,  p-value: 4.81e-06

plot(growth$muMC,growth$relincrease,pch=c(21,22,24)[as.factor(growth$Light)],bg=c(CMcol,HPcol,RScol,SMcol))

## Warning in arrows(C.summ[, 5], I.summ[, 5] + I.summ$se, C.summ[, 5],
## I.summ[, : zero-length arrow is of indeterminate angle and so skipped

#legend(x = -0.15, y = 12, legend = c('0 uE', '50 uE', '100 uE', 'CM', 'HP', 'RS', 'SM'),cex=ptcex,pt.cex=1.5
```

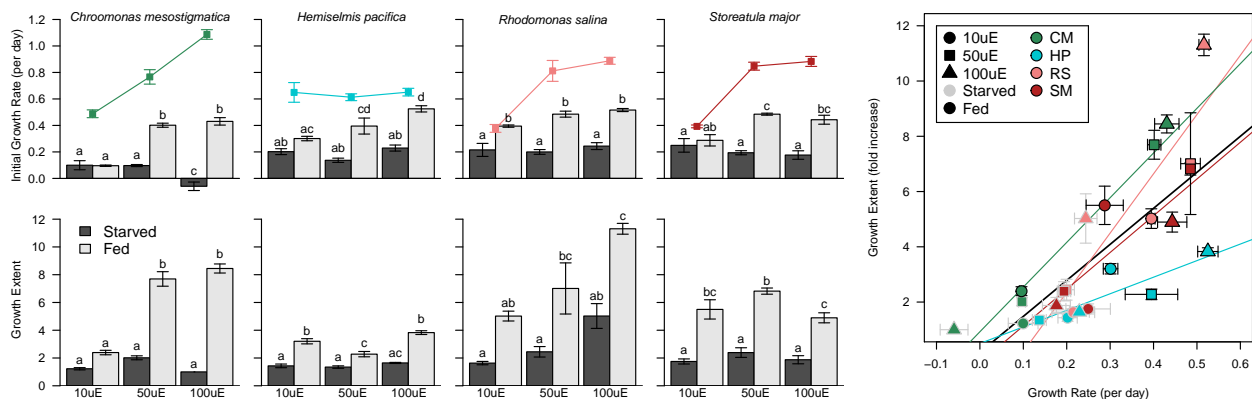


Figure 5: Heterotrophic contributions to growth

```
#layout(matrix(c(1,1,1,2,2,2,8,3,3,3,4,4,4,7,5,5,5,6,6,6,9),nrow=(6*3+3),ncol=1))
layout(matrix(c(1,1,1,2,2,3,3,3,4,4),nrow=2,ncol=5,byrow=TRUE))
xcoords <- c(1,2,3,4.5,5.5,6.5,8,9,10,11.5,12.5,13.5)

# ingestion rate
par(mar=c(2,4,0.5,0.5))
bargraph.CI(Prey,ingest_2,group=Light,data=growth[growth$Fed=='Y',],legend=FALSE,x.leg=1,y.leg=15,ylim=c(0,10))

par(mar=c(4,4,1,0.5))

# SECOND PLOT: ingestion rate vs growth rate
lm.all <- lm(growth[growth$Fed=='Y',]$muMC~growth[growth$Fed=='Y',]$ingest_2); summary(lm.all) #signif.

##
## Call:
## lm(formula = growth[growth$Fed == "Y", ]$muMC ~ growth[growth$Fed ==
##      "Y", ]$ingest_2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.27035 -0.04367  0.01011  0.07994  0.16933
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.266463    0.042546   6.263 3.93e-07
## growth[growth$Fed == "Y", ]$ingest_2 0.017805    0.005259   3.386  0.0018
##
## (Intercept)          ***
## growth[growth$Fed == "Y", ]$ingest_2 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1089 on 34 degrees of freedom
## Multiple R-squared:  0.2522, Adjusted R-squared:  0.2302
## F-statistic: 11.46 on 1 and 34 DF, p-value: 0.001805

lm.RS <- lm(muMC~ingest_2,data=growth[growth$Prey=='RS'&growth$Fed=='Y',]); summary(lm.RS) #signif.; F-

##
## Call:
## lm(formula = muMC ~ ingest_2, data = growth[growth$Prey == "RS" &
##      growth$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.085170 -0.056651  0.008239  0.059667  0.068218
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.459887    0.081843   5.619  0.0008 ***
```

```

## ingest_2    0.000672    0.009708    0.069    0.9468
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06338 on 7 degrees of freedom
## Multiple R-squared:  0.000684, Adjusted R-squared:  -0.1421
## F-statistic: 0.004791 on 1 and 7 DF, p-value: 0.9468
lm.SM <- lm(muMC~ingest_2,data=growth[growth$Prey=='SM'&growth$Fed=='Y',]); summary(lm.SM) #signif.; F-

##
## Call:
## lm(formula = muMC ~ ingest_2, data = growth[growth$Prey == "SM" &
##      growth$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.07256 -0.05838  0.00850  0.05436  0.08521
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.189548   0.067162   2.822   0.0257 *
## ingest_2     0.020717   0.006088   3.403   0.0114 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06686 on 7 degrees of freedom
## Multiple R-squared:  0.6233, Adjusted R-squared:  0.5694
## F-statistic: 11.58 on 1 and 7 DF, p-value: 0.0114
lm.CM <- lm(muMC~ingest_2,data=growth[growth$Prey=='CM'&growth$Fed=='Y',]); summary(lm.CM) #signif.; F-

##
## Call:
## lm(formula = muMC ~ ingest_2, data = growth[growth$Prey == "CM" &
##      growth$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.21576 -0.15231  0.01321  0.11083  0.16924
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.18688   0.11165   1.674   0.138
## ingest_2     0.02215   0.01782   1.243   0.254
##
## Residual standard error: 0.1578 on 7 degrees of freedom
## Multiple R-squared:  0.1808, Adjusted R-squared:  0.06382
## F-statistic: 1.545 on 1 and 7 DF, p-value: 0.2538
lm.HP <- lm(muMC~ingest_2,data=growth[growth$Prey=='HP'&growth$Fed=='Y',]); summary(lm.HP) #signif.; F-

##
## Call:
## lm(formula = muMC ~ ingest_2, data = growth[growth$Prey == "HP" &
##      growth$Fed == "Y", ])

```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.094714 -0.028945  0.006322  0.040034  0.071321
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.12162    0.06364   1.911  0.09763 .
## ingest_2      0.05503    0.01166   4.720  0.00216 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0593 on 7 degrees of freedom
## Multiple R-squared:  0.7609, Adjusted R-squared:  0.7267
## F-statistic: 22.27 on 1 and 7 DF,  p-value: 0.002159
plot(growth[growth$Fed=='Y'],$ingest_2,growth[growth$Fed=='Y'],$muMC,las=1,pch=c(21,22,24)[as.factor(growth$Prey)],
C.summ <- summarySE(data=growth[growth$Fed=='Y'],measurevar='muMC',groupvars=c('Light','Prey')); I.summary(C.summ)
legend(13.1,0.29,legend=c('10uE','50uE','100uE'),pch=c(21,22,24),pt.cex=ptcex,pt.bg='black',col='black')

# linear scale: prey per initial MC
par(mar=c(2,4,0.5,0.5))
bargraph.CI(Prey,(PreyperMC),group=Light,data=growth[growth$Fed=='Y'],legend=FALSE,err.width=0.05,names=c('Light','Prey'))

# FOURTH PLOT: total ingestion vs growth extent
par(mar=c(4,4,1,0.5))
growth.y <- growth[growth$Fed=='Y',]
lm.all <- lm(growth.y$relincrease~log(growth.y$PreyperMC)); summary(lm.all) #signif.; F-stat = 8.4, p-value = 0.006525

##
## Call:
## lm(formula = growth.y$relincrease ~ log(growth.y$PreyperMC))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.4740 -1.2625 -0.4977  0.6485  6.5826
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.6685     1.7857   0.374  0.71047
## log(growth.y$PreyperMC)  1.4408     0.4971   2.898  0.00652 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.511 on 34 degrees of freedom
## Multiple R-squared:  0.1981, Adjusted R-squared:  0.1745
## F-statistic: 8.4 on 1 and 34 DF,  p-value: 0.006525
```

```

lm.RS.f <- lm(relincrease~log(PreyperMC),data=growth.y[growth.y$Prey=='RS'&growth.y$Fed=='Y',]); summary

##
## Call:
## lm(formula = relincrease ~ log(PreyperMC), data = growth.y[growth.y$Prey ==
## "RS" & growth.y$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.0784 -1.4995 -0.8877  2.9664  4.1790
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -9.476     11.599  -0.817   0.441
## log(PreyperMC)   5.606      3.754   1.493   0.179
##
## Residual standard error: 3.016 on 7 degrees of freedom
## Multiple R-squared:  0.2416, Adjusted R-squared:  0.1333
## F-statistic:  2.23 on 1 and 7 DF,  p-value: 0.179

lm.SM.f <- lm(relincrease~log(PreyperMC),data=growth.y[growth.y$Prey=='SM'&growth.y$Fed=='Y',]); summary

##
## Call:
## lm(formula = relincrease ~ log(PreyperMC), data = growth.y[growth.y$Prey ==
## "SM" & growth.y$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.45464 -0.81082  0.05431  0.65520  1.45024
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.056      4.534   0.233   0.823
## log(PreyperMC)   1.380      1.332   1.036   0.335
##
## Residual standard error: 1.1 on 7 degrees of freedom
## Multiple R-squared:  0.133, Adjusted R-squared:  0.009104
## F-statistic: 1.073 on 1 and 7 DF,  p-value: 0.3346

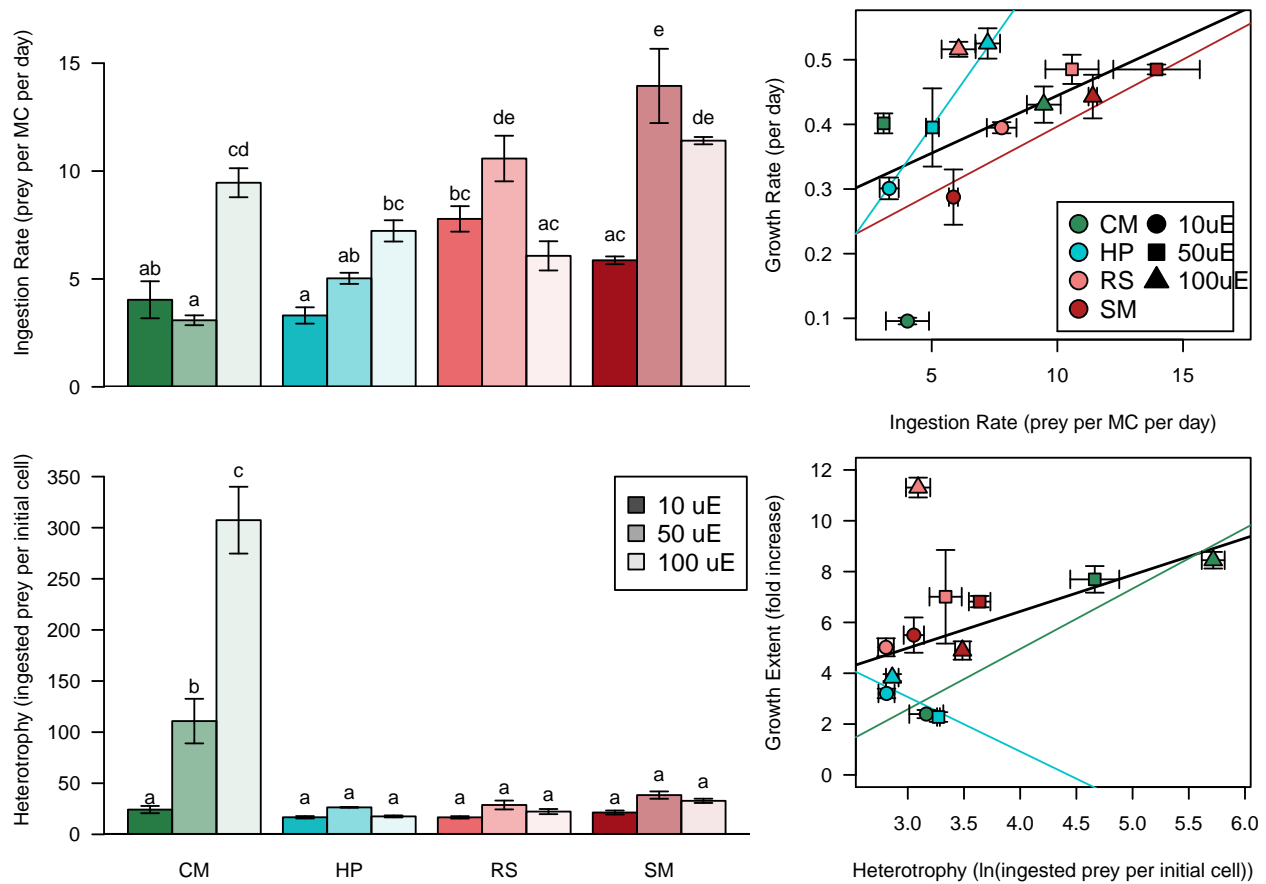
lm.CM.f <- lm(relincrease~log(PreyperMC),data=growth.y[growth.y$Prey=='CM'&growth.y$Fed=='Y',]); summary

##
## Call:
## lm(formula = relincrease ~ log(PreyperMC), data = growth.y[growth.y$Prey ==
## "CM" & growth.y$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.96250 -0.71021 -0.18499 -0.08223  1.92060
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -4.5418     1.6693  -2.721 0.029730 *
## log(PreyperMC)   2.3742     0.3597   6.601 0.000304 ***

```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.158 on 7 degrees of freedom
## Multiple R-squared:  0.8616, Adjusted R-squared:  0.8418
## F-statistic: 43.57 on 1 and 7 DF,  p-value: 0.0003041
lm.HP.f <- lm(relincrease~log(PreyperMC),data=growth.y[growth.y$Prey=='HP'&growth.y$Fed=='Y',]); summary(lm.HP.f)

##
## Call:
## lm(formula = relincrease ~ log(PreyperMC), data = growth.y[growth.y$Prey ==
##      "HP" & growth.y$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.8070 -0.2609 -0.0068  0.2690  0.7411
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      9.4956      2.5353   3.745  0.00721 **
## log(PreyperMC)  -2.1434      0.8479  -2.528  0.03936 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5596 on 7 degrees of freedom
## Multiple R-squared:  0.4772, Adjusted R-squared:  0.4025
## F-statistic:  6.39 on 1 and 7 DF,  p-value: 0.03936
plot(log(as.numeric(as.character(growth.y$PreyperMC))),growth.y$relincrease,pch=c(21,22,24)[as.factor(growth.y$Prey)],main="log(PreyperMC) vs relincrease")
```

#legend(1050,12.2,legend=c('CM','HP','RS','SM'),pch=21,pt.cex=pt.cex,pt.bg=preycolvec,bty='n'); arrows(c

Figure 6: Photosynthetic contributions to growth

```
layout(matrix(c(8,1,2,3,4,rep(7,3),9,5,5,5,5,rep(7,3),10,6,6,6,6,rep(7,3)),nrow=3,ncol=8,byrow=TRUE));

growth$C.fixed3 <- growth$C.fixed2/1000 # convert from pg to ng
growth$C.per.prey <- growth$C.fixed2/growth$TotPreyCons

xcoords <- c(1.2,3.4,5.6)
xcoords2 <- c(.7,1.7,2.9,3.9,5.1,6.1)
#quartz(height=3.5,width=7.5)
par(mar=c(1,1,0.5,0.5))
#layout(matrix(c(7,1,1,1,2,2,3,3,3,4,4,4,rep(6,6),8,rep(5,3*4),rep(6,6)),nrow=2,ncol=19,byrow=TRUE))

# Row 1: Photosynthate
ymax <- 50; ymin <- 0; bargraph.CI(Light,C.fixed3,group=Fed,data=growth[growth$Prey=='CM'],las=1,legend=FALSE)

bargraph.CI(Light,C.fixed3,group=Fed,data=growth[growth$Prey=='HP'],las=1,legend=FALSE,leg.lab=c('Star'))

bargraph.CI(Light,C.fixed3,group=Fed,data=growth[growth$Prey=='RS'],las=1,legend=FALSE,ylim=c(ymin,ymax))
```

```

bargraph.CI(Light,C.fixed3,group=Fed,data=growth[growth$Prey=='SM',],las=1,legend=TRUE,leg.lab=c('Starv

# Second row: fold difference
xcoords <- c(1,2,3,4.5,5.5,6.5,8,9,10,11.5,12.5,13.5)

bargraph.CI(Prey,C.fixed2/abs(deltaCfixed-C.fixed2),group=Light,data=growth[growth$Fed=='Y',],legend=FALSE,

#TukeyHSD(aov(C.fixed2/abs(deltaCfixed-C.fixed2)~as.factor(Light)*Prey,data=growth[growth$Fed=='Y',]))

# Third row: Carbon fixed per prey ingested
bargraph.CI(Prey,C.per.prey,group=Light,data=growth[growth$Fed=='Y',],legend=FALSE,err.width=0.05,las=3

#TukeyHSD(aov(C.per.prey~as.factor(Light)*Prey,data=growth[growth$Fed=='Y',]))

# FOURTH PLOT: total photosynthate vs growth extent
par(mar=c(4,5,.5,0.5))

lm.all <- lm(growth$relincrease~growth$C.fixed3); summary(lm.all) #signif.; F-stat = 363.6, p-val = < 2

##
## Call:
## lm(formula = growth$relincrease ~ growth$C.fixed3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.7400 -0.7034 -0.2552  0.6855  3.4925
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.46273    0.18230   8.024 1.65e-11 ***
## growth$C.fixed3 0.19933    0.01041  19.152 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.132 on 70 degrees of freedom
## Multiple R-squared:  0.8397, Adjusted R-squared:  0.8374
## F-statistic: 366.8 on 1 and 70 DF,  p-value: < 2.2e-16
lm.RS.f <- lm(relincrease~C.fixed3,data=growth[growth$Prey=='RS'&growth$Fed=='Y',]); summary(lm.RS.f) #

##
## Call:
## lm(formula = relinincrease ~ C.fixed3, data = growth[growth$Prey ==
##      "RS" & growth$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.58623 -0.24859  0.08797  0.55830  1.53320
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.59415    0.75984   3.414 0.011225 *

```

```

## C.fixed3      0.20795      0.02661      7.814 0.000106 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.111 on 7 degrees of freedom
## Multiple R-squared:  0.8972, Adjusted R-squared:  0.8825
## F-statistic: 61.06 on 1 and 7 DF,  p-value: 0.0001059
lm.RS.n <- lm(relincrease~C.fixed3,data=growth[growth$Prey=='RS'&growth$Fed=='N',]); summary(lm.RS.n) #

##
## Call:
## lm(formula = relincrease ~ C.fixed3, data = growth[growth$Prey ==
##      "RS" & growth$Fed == "N", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.60262 -0.35354 -0.06898  0.27792  0.72256
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.19688     0.25227   4.744  0.0021 **
## C.fixed3      0.17056     0.01769   9.640 2.72e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4955 on 7 degrees of freedom
## Multiple R-squared:  0.9299, Adjusted R-squared:  0.9199
## F-statistic: 92.92 on 1 and 7 DF,  p-value: 2.724e-05
lm.SM.f <- lm(relincrease~C.fixed3,data=growth[growth$Prey=='SM'&growth$Fed=='Y',]); summary(lm.SM.f) #

##
## Call:
## lm(formula = relincrease ~ C.fixed3, data = growth[growth$Prey ==
##      "SM" & growth$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.40307 -1.20669  0.05384  0.96935  1.36544
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.36265     0.95182   5.634 0.000787 ***
## C.fixed3      0.02059     0.04768   0.432 0.678818
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.166 on 7 degrees of freedom
## Multiple R-squared:  0.02595, Adjusted R-squared: -0.1132
## F-statistic: 0.1865 on 1 and 7 DF,  p-value: 0.6788
lm.SM.n <- lm(relincrease~C.fixed3,data=growth[growth$Prey=='SM'&growth$Fed=='N',]); summary(lm.SM.n) #

##
## Call:

```

```

## lm(formula = relincrease ~ C.fixed3, data = growth[growth$Prey ==
##      "SM" & growth$Fed == "N", ])
##
## Residuals:
##      Min        1Q    Median        3Q        Max
## -0.62041 -0.35385 -0.01644  0.24170  0.87923
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.70731     0.31843   5.362  0.00105 **
## C.fixed3      0.04822     0.04356   1.107  0.30491
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5083 on 7 degrees of freedom
## Multiple R-squared:  0.149, Adjusted R-squared:  0.02739
## F-statistic: 1.225 on 1 and 7 DF, p-value: 0.3049
lm.CM.f <- lm(relincrease~C.fixed3,data=growth[growth$Prey=='CM'&growth$Fed=='Y',]); summary(lm.CM.f) #

##
## Call:
## lm(formula = relincrease ~ C.fixed3, data = growth[growth$Prey ==
##      "CM" & growth$Fed == "Y", ])
##
## Residuals:
##      Min        1Q    Median        3Q        Max
## -0.55407 -0.29453  0.04931  0.17498  0.76372
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.245605     0.241095   9.314 3.41e-05 ***
## C.fixed3      0.156242     0.007878  19.832 2.07e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4114 on 7 degrees of freedom
## Multiple R-squared:  0.9825, Adjusted R-squared:  0.98
## F-statistic: 393.3 on 1 and 7 DF, p-value: 2.072e-07
lm.CM.n <- lm(relincrease~C.fixed3,data=growth[growth$Prey=='CM'&growth$Fed=='N',]); summary(lm.CM.n) #

##
## Call:
## lm(formula = relincrease ~ C.fixed3, data = growth[growth$Prey ==
##      "CM" & growth$Fed == "N", ])
##
## Residuals:
##      Min        1Q    Median        3Q        Max
## -0.5348 -0.5008  0.1913  0.2208  0.5839
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.9822     0.3281   2.994  0.0201 *
## C.fixed3       0.2358     0.1594   1.480  0.1825

```

```

## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4506 on 7 degrees of freedom
## Multiple R-squared:  0.2383, Adjusted R-squared:  0.1294
## F-statistic:  2.19 on 1 and 7 DF,  p-value: 0.1825
lm.HP.f <- lm(relincrease~C.fixed3,data=growth[growth$Prey=='HP'&growth$Fed=='Y',]); summary(lm.HP.f) #

##
## Call:
## lm(formula = relinincrease ~ C.fixed3, data = growth[growth$Prey ==
##      "HP" & growth$Fed == "Y", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.86859 -0.12767 -0.00792  0.28699  0.74476
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.65699    0.46685   3.549  0.00935 **
## C.fixed3      0.26614    0.08058   3.303  0.01307 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4839 on 7 degrees of freedom
## Multiple R-squared:  0.6091, Adjusted R-squared:  0.5533
## F-statistic: 10.91 on 1 and 7 DF,  p-value: 0.01307
lm.HP.n <- lm(relincrease~C.fixed3,data=growth[growth$Prey=='HP'&growth$Fed=='N',]); summary(lm.HP.n) #

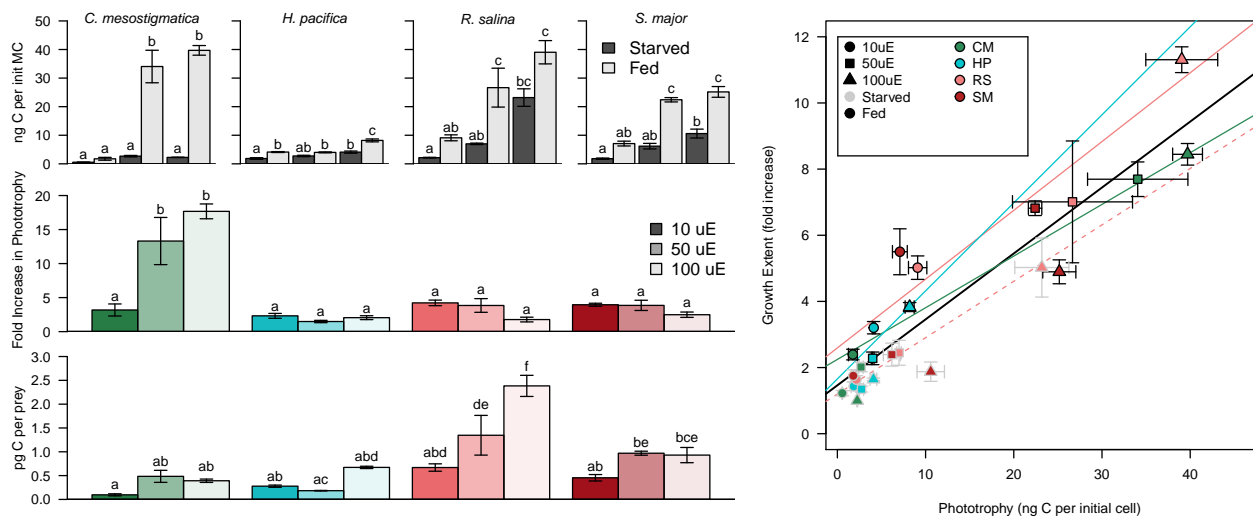
##
## Call:
## lm(formula = relinincrease ~ C.fixed3, data = growth[growth$Prey ==
##      "HP" & growth$Fed == "N", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.225199 -0.113317  0.005071  0.097854  0.232258
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.1518    0.1637   7.035 0.000205 ***
## C.fixed3      0.1117    0.0533   2.095 0.074397 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1625 on 7 degrees of freedom
## Multiple R-squared:  0.3854, Adjusted R-squared:  0.2976
## F-statistic:  4.39 on 1 and 7 DF,  p-value: 0.0744
plot(growth$C.fixed2/1000,growth$relincrease,pch=c(21,22)[growth$Fed],bg=preycolvec[growth$Prey],las=1,

## Warning in arrows(C.summ[, 5]/1000, I.summ[, 5] + I.summ$sse, C.summ[, 5]/
## 1000, : zero-length arrow is of indeterminate angle and so skipped

```

```
TukeyHSD(aov(C.fixed3~Fed*as.factor(Light),data=growth[growth$Prey=='SM',]))
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = C.fixed3 ~ Fed * as.factor(Light), data = growth[growth$Prey == "SM", ])
##
## $Fed
##      diff      lwr      upr p adj
## Y-N 12.03204 9.951656 14.11241 0
##
## $`as.factor(Light)`
##      diff      lwr      upr      p adj
## 50-10  9.852058 6.7322214 12.971895 0.0000061
## 100-10 13.429425 10.3095887 16.549262 0.0000002
## 100-50  3.577367 0.4575305 6.697204 0.0249375
##
## $`Fed:as.factor(Light)`
##      diff      lwr      upr      p adj
## Y:10-N:10  5.2820940 -0.2728926 10.837081 0.0658299
## N:50-N:10  4.3741001 -1.1808865 9.929087 0.1592640
## Y:50-N:10 20.6121102 15.0571236 26.167097 0.0000004
## N:100-N:10  8.7824715 3.2274850 14.337458 0.0019704
## Y:100-N:10 23.3584733 17.8034867 28.913460 0.0000001
## N:50-Y:10 -0.9079939 -6.4629805 4.646993 0.9926408
## Y:50-Y:10 15.3300162 9.7750296 20.885003 0.0000094
## N:100-Y:10  3.5003776 -2.0546090 9.055364 0.3407546
## Y:100-Y:10 18.0763793 12.5213927 23.631366 0.0000016
## Y:50-N:50 16.2380101 10.6830236 21.792997 0.0000051
## N:100-N:50  4.4083715 -1.1466151 9.963358 0.1542413
## Y:100-N:50 18.9843732 13.4293867 24.539360 0.0000009
## N:100-Y:50 -11.8296386 -17.3846252 -6.274652 0.0001313
## Y:100-Y:50  2.7463631 -2.8086235 8.301350 0.5786171
## Y:100-N:100 14.5760018 9.0210152 20.130988 0.0000160
```



Supplementary Figures: Timeseries of cells and pigments

```

# Choose the data columns to use, their names, and their maxima
datcols <- c(8,9,30,dim(dat)[2]-2,dim(dat)[2]-1) # data columns to use. CHECK THIS CAREFULLY
ylabstrs <- c('MC (cells/mL)', 'Prey (cells/mL)', 'Plastids (per MC cell)', 'MC chl (pg/cell)', 'Tot. chl (pg/cell)')
ymax <- c(10000,25000,30,55,100)

# We will partition data by light level
lightvec <- c(10,50,100)
titlestr <- paste(as.character(lightvec), 'uE'); titlestr <- rbind(titlestr, c(rep('',4)), c(rep('',4)), c(rep('',4)))

# Iterate over different prey types
preyvec <- c('CM', 'HP', 'RS', 'SM')
preycolvec <- c(CMcol, HPcol, RScol, SMcol)

# Step 1: choose your prey type & subset data
for(k in 1:length(preyvec)){

# Step 1: Choose the prey and subset data accordingly
preychoice <- preyvec[k]
dat.hold2 <- dat[dat$Prey==preychoice,]

par(mar=c(2,4,2,1),mfcol=c(length(datcols),length(lightvec)))
# Note that we've used the argument "mfcol" instead of "mfrow." This ensures that the subsequent mega-plots are in columns

for(j in 1:length(lightvec)){

# Step 2: Subset the data again for appropriate light level
dat.hold <- dat.hold2[dat.hold2$Light==lightvec[j],]
# Step 3: Create summary stats.
summ.MC<-summarySE(data=dat.hold,measurevar="MC_cellspmL",groupvars=c("ExptDay", 'Fed'),na.rm=TRUE)
summ.prey<-summarySE(data=dat.hold,measurevar="Prey_cellspmL",groupvars=c("ExptDay", 'Fed'),na.rm=TRUE)
summ.chl<-summarySE(data=dat.hold,measurevar="chl_p_cell",groupvars=c("ExptDay", 'Fed'),na.rm=TRUE)
summ.chl.p.mL <- summarySE(data=dat.hold,measurevar='chl_p_mL',groupvars=c("ExptDay", 'Fed'),na.rm=TRUE)
summ.Plastids<-summarySE(data=dat.hold,measurevar="PlastidCount",groupvars=c("ExptDay", 'Fed'),na.rm=TRUE)

# Step 4: Make all the plots
for(i in 1:length(datcols)){

# Create axes using raw data
plot(dat.hold$ExptDay, dat.hold[,datcols[i]], type='n', las=1, ylab=ylabstrs[i], xlab='', main=titlestr[i])

if(i == 1){ summ.use <- summ.MC }; if(i == 2){ summ.use <- summ.prey }; if(i == 4){ summ.use <- summ.chl.p.mL }; if(i == 5){ summ.use <- summ.Plastids }

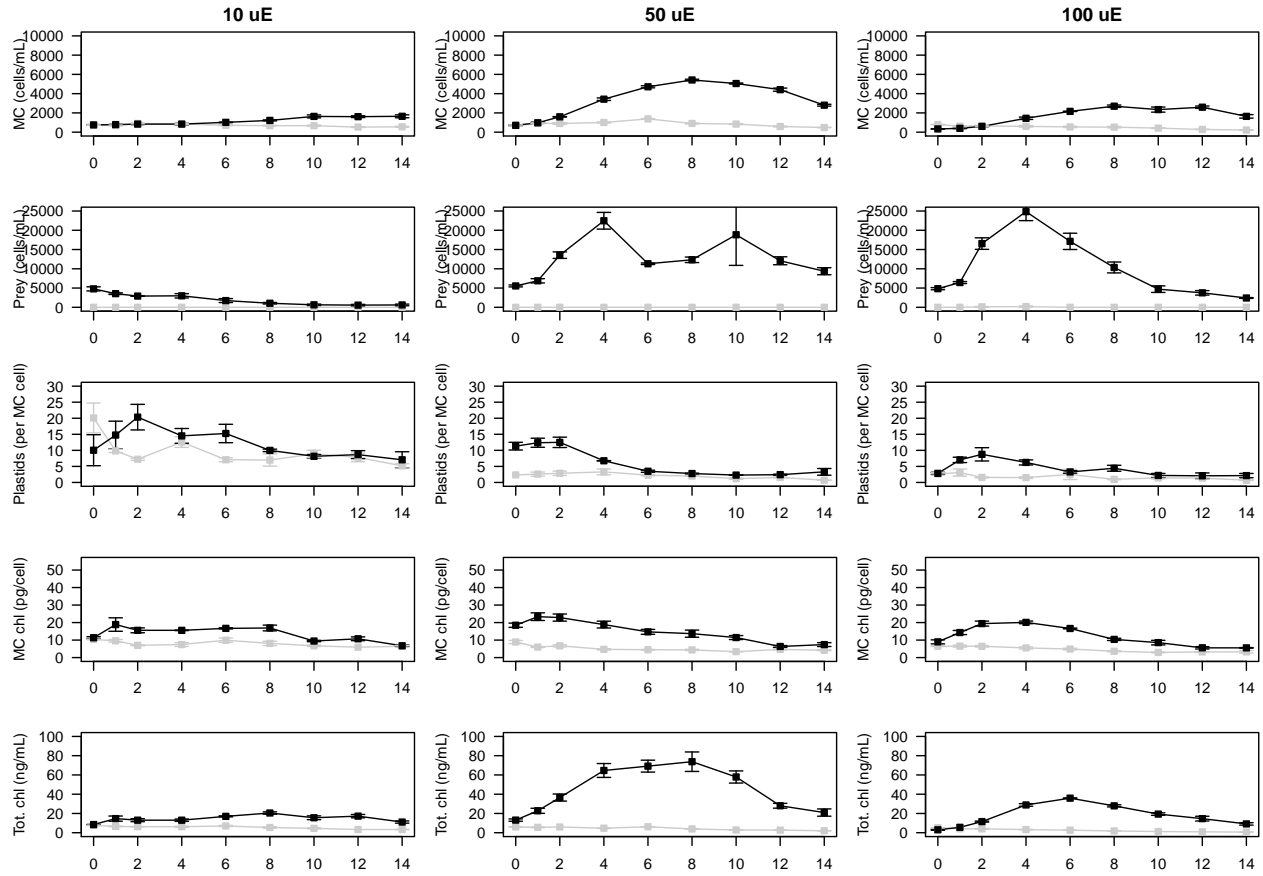
# Plot lines connecting means
lines(summ.use[summ.use$Fed=='N',]$ExptDay,summ.use[summ.use$Fed=='N',][,4],col= Starvedcol); lines(summ.use[summ.use$Fed=='Y',]$ExptDay,summ.use[summ.use$Fed=='Y',][,4],col= Fedcol)

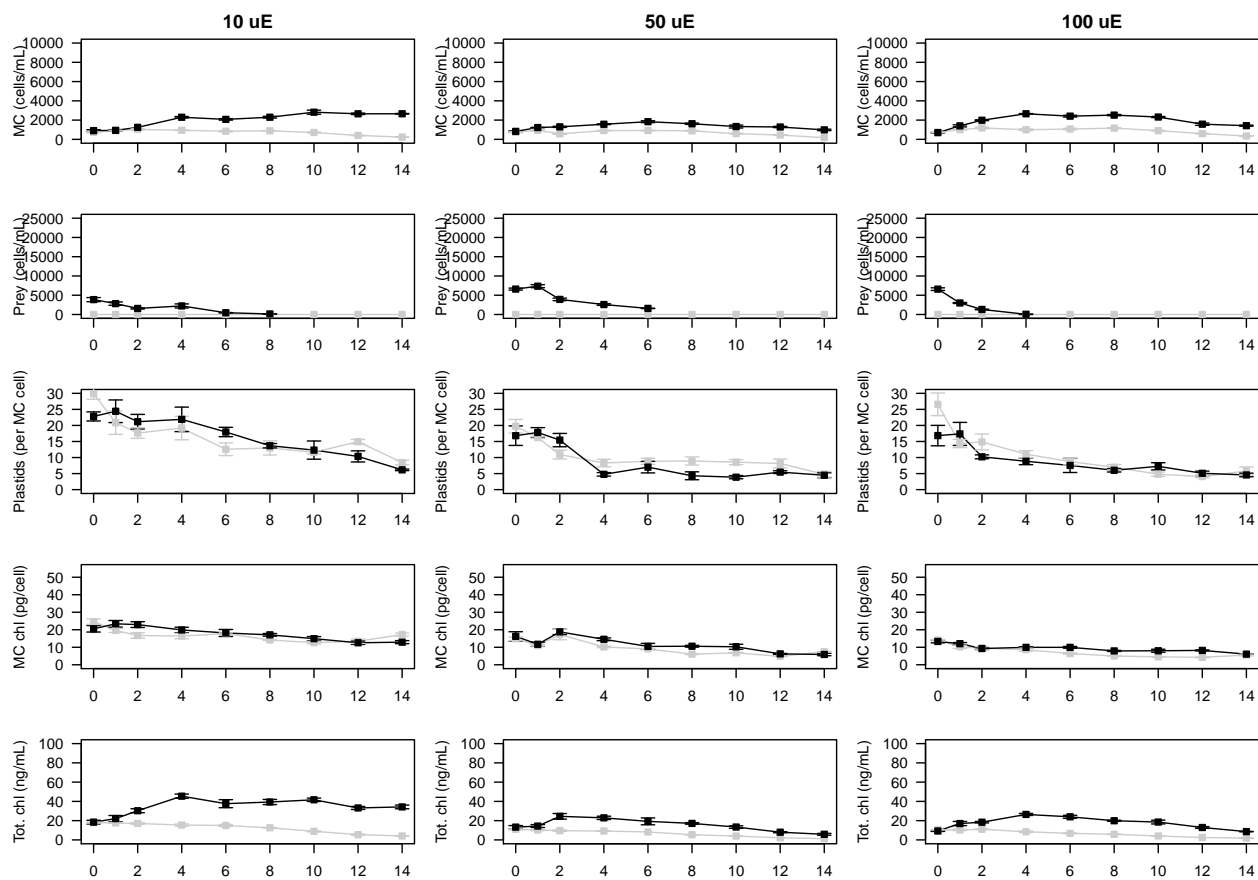
# Plot std error
summ.use.Y <- summ.use[summ.use$Fed=="Y",]; summ.use.N <- summ.use[summ.use$Fed=="N",]
arrows(summ.use.Y$ExptDay,summ.use.Y[,4]+summ.use.Y$se,summ.use.Y$ExptDay,summ.use.Y[,4]-summ.use.Y$se,angle=90)
}
}

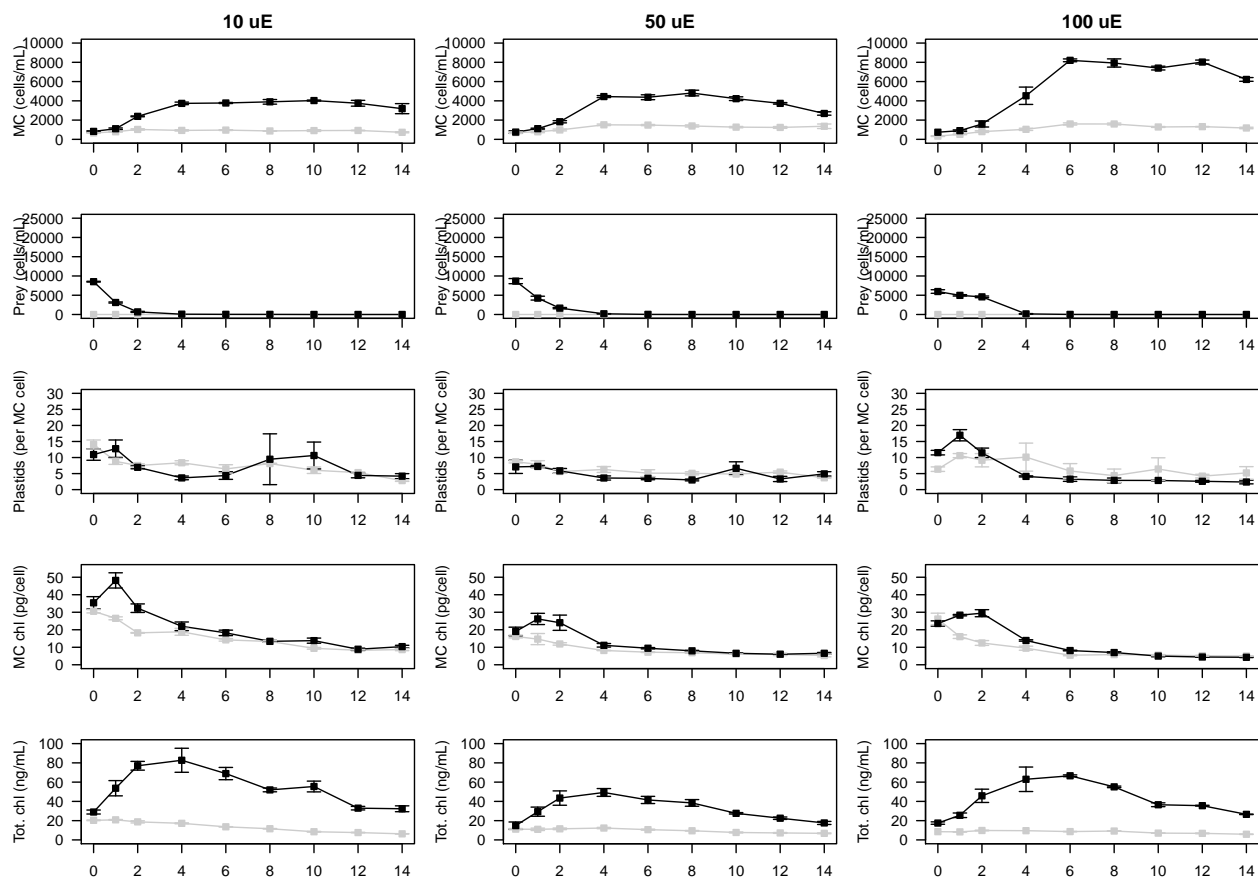
```

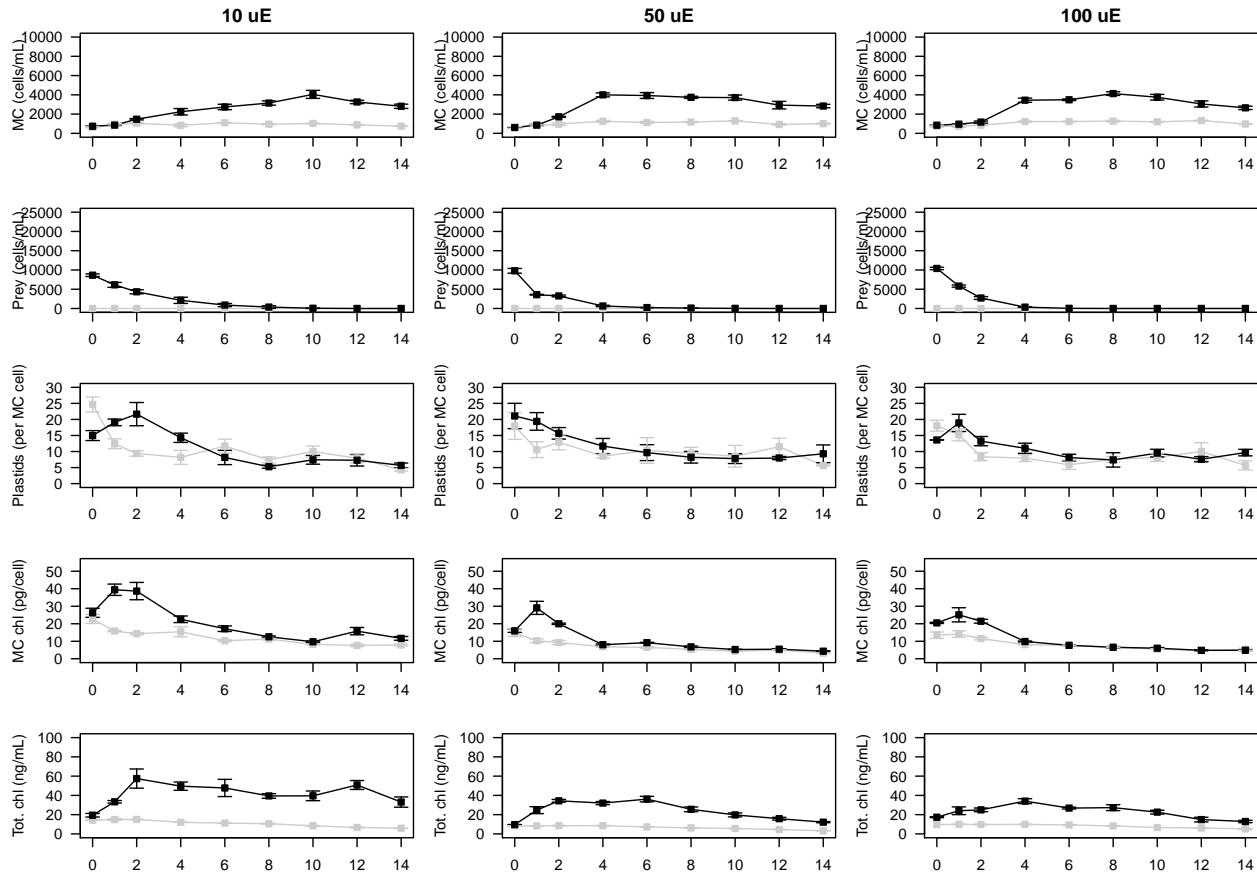
```
# Plot means
```

```
points(summ.use$ExptDay, summ.use[,4], pch=22, col=c(Starvedcol, Fedcol)[summ.use$Fed], bg=c(Starvedcol, Fedcol))
}}
}
```









Supplementary Figures: Photophysiology timeseries

```
# Choose the data columns to use, their names, and their maxima
datcols <- c(12,21,dim(dat)[2]-4,dim(dat)[2]-3) # data columns to use. CHECK THIS CAREFULLY
ylabstrs <- c('Photosynth. Eff. (Fv/Fm)', 'ETR', 'PI_chl', 'PI_cell') # y-labels
ymax <- rbind(rep(0.6,3),c(20,80,150),c(2,5,10),c(30,100,150))
ymin <- c(0.1,0,0,0)

# We will partition data by light level
lightvec <- c(10,50,100)
titlestr <- paste(as.character(lightvec),'uE'); titlestr <- rbind(titlestr,c(rep('',4)),c(rep('',4)),c(rep('',4)))

# Iterate over different prey types
preyvec <- c('CM','HP','RS','SM')

# Step 1: choose your prey type & subset data
for(k in 1:length(preyvec)){

# Step 1: Choose the prey and subset data accordingly
preychoice <- preyvec[k]
dat.hold2 <- dat[dat$Prey==preychoice,]
crypto.hold2 <- crypto.dat[crypto.dat$Prey==preychoice,]
```

```

par(mar=c(2,4,2,1),mfcol=c(length(datcols),length(lightvec)))
# Note that we've used the argument "mfcol" instead of "mfrow." This ensures that the subsequent mega-p

for(j in 1:length(lightvec)){

# Step 2: Subset the data again for appropriate light level
dat.hold <- dat.hold2[dat.hold2$Light==lightvec[j],]
crypto.hold <-crypto.hold2[crypto.hold2$Light==lightvec[j],]
# Step 3: Create summary stats.
summ.FvFm<-summarySE(data=dat.hold,measurevar="FvFm",groupvars=c("ExptDay", 'Fed'),na.rm=TRUE); crypto.F
summ.ETR <- summarySE(data=dat.hold,measurevar='ETR_I',groupvars=c("ExptDay", 'Fed'),na.rm=TRUE); crypto

summ.PIchl<-summarySE(data=dat.hold,measurevar="PI_chl",groupvars=c("ExptDay", 'Fed'),na.rm=TRUE); crypto
summ.PIcell<-summarySE(data=dat.hold,measurevar="PI_cell",groupvars=c("ExptDay", 'Fed'),na.rm=TRUE); cryp

# Step 4: Make all the plots
for(i in 1:length(datcols)){

# Create axes using raw data
plot(dat.hold$ExptDay, dat.hold[,datcols[i]], type='n', las=1, ylab=ylabstrs[i], xlab='', main=titlestr

if(i == 1){ summ.use <- summ.FvFm; crypto.use<-crypto.FvFm }; if(i == 2){ summ.use <- summ.ETR; crypto.

# Plot lines connecting means
lines(summ.use[summ.use$Fed=='N'],)$ExptDay,summ.use[summ.use$Fed=='N'],[,4],col= Starvedcol); lines(summ

# Plot std error
summ.use.Y <- summ.use[summ.use$Fed=="Y",]; summ.use.N <- summ.use[summ.use$Fed=="N",]
arrows(summ.use.Y$ExptDay,summ.use.Y[,4]+summ.use.Y$se,summ.use.Y$ExptDay,summ.use.Y[,4]-summ.use.Y$se,

# Plot means
points(summ.use$ExptDay,summ.use[,4],pch=22,col=c(Starvedcol,Fedcol)[summ.use$Fed],bg=c(Starvedcol,Fedcol

}}
}

```

